Opera » KDE 4.1 » DjVu » Сисадминам

LXFDVD

Mythbuntu

Домашний медиацентр

ПЛЮС: VirtualBox
» Дистрибутивы будущего
» ALT Linux 4.1 Desktop

LINUX FORMAT

Главное в мире Linux

Ноябрь 2008 № 11 (111)

30 дней в Haiku
Linux – не единственный вариант с. 48

НАШ ОТВЕТ: MythTV

Записывайте телепрограммы, гуляйте по Сети, смотрите фотографии и играйте, не вставая с дивана!

Компиляция 911
Собирайте программы самостоятельно с. 44

Nagios
Наблюдай за сетью и властвуй над ней с. 78

Самоорганизация
Синхронизируем календарь с мобильником с. 74

Двойная загрузка
Слишком привыкли к Vista? Пользуйтесь благами двух цивилизаций с. 64

«Я всегда говорю людям: если вам не в кайф работать с Cairo, значит, что-то неправильно.»

Карл Ворт. Разработчик Cairo с. 28

Каталог агентства «РОСПЕЧАТЬ» – подписной индекс 20882
Каталог «ПРЕССА РОССИИ» – подписной индекс 87974

Linux Center
www.linuxcenter.ru

LINUX FORMAT
Главное в мире Linux

Счастье вашего очага

Мы-то знаем, что это просто круто, но другим могут потребоваться какие-то аргументы. В общем, сегодня команда LXF дает ответ на вопрос: «Зачем нужен Linux-компьютер в вашей гостиной?»

Грэм Моррисон
Шум машины поможет заглушить голоса в голове. Эй, на мне это сработало!

Майк Сондерс
Процедура швартовки во Frontier: *Elite 2* заслуживает 50-дюймового экрана и звука 5.1.

Ник Вейч
В моей квартирке нет места для Linux, поэтому я использую MikeOS: она куда меньше.

Эфрейн Эрнандес-Мендоса
Если вы начнете настраивать MythTV прямо сейчас, есть неплохой шанс успеть закончить к ЧМ-2010.

Эндрью Грегори
Простите, тут я вам не советчик. Пока компьютеры не начнут выпускать в зеленом кожаном корпусе, они не будут сочетаться с моими занавесками.

Энди Ченнел
Я уже поставил Mythbuntu в гостиной, Ubuntu в спальне и Poobuntu – в туалете.

Дэвид Картрайт
Я смотрю на это по-другому: я живу в серверной. Сухо, тепло и можно без помех слушать Radio 4.

Д-р Крис Браун
А что, сейчас не 2008 год? Так разве еще остались люди, у которых нет Linux в гостиной?

Майкл Дж Хэммел
Как и многим GIMP-художникам, мне нужна своя мастерская. Почему бы не организовать ее поближе к дивану?

Саймон Пикстон
Можно не вставать с кресла, играя в *Eve Online*. И иметь выход в Интернет в двух шагах от холодильника.

Джульетта Кемп
Полезно, чтобы машина всегда была под присмотром (да, я параноик).

Сьюзан Линтон
Если цены на тепло продолжат расти, я перенесу в гостиную все компьютеры и буду там греться.

Миссия журнала

- Пропаганда свободного ПО в России
- Продвижение решений с открытым кодом в бизнес-сообществе
- Поддержка российского Open Source сообщества
- Организация трибуны для разработчиков свободного ПО
- Обратная связь между разработчиками и потребителями ПО

30 дней в...

...Haiku. Так называется одна из статей в этом номере журнала. Грэм Моррисон, ни разу до того не видевший BeOS (кстати, жаль – хорошая была система) попробовал спрогнозировать, что произойдет, если установить эту незнакомую и, строго говоря, не предназначенную для повседневного использования ОС, на ваш персональный компьютер. Его размышления можно прочесть на стр. 48, я же хочу поговорить о другом. Статьи, подобные этой, выходили и три, и пять, и десять лет назад, только тогда в роли Haiku выступал Linux. Смельчаки, движимые редакционным заданием, а то и собственным любопытством, ставили какой-нибудь Black Cat второй системой (виртуализация еще не была столь распространена) и глядели, что из этого выйдет (и выйдет ли хоть что-нибудь).

Что мы имеем сегодня? Вопрос о готовности Linux к использованию человеком с небольшим опытом – в нашем случае подружкой (стр. 40) – по-прежнему не снят с повестки дня, но месяц один на один с этой системой уже едва ли кажется борьбой за выживание. Что там тридцать дней – у иных Linux-ПК uptime больше, чем все время существования на рынке Windows Vista! То, чего мы так долго ждали, свершилось: из категории экспериментальных Linux плавно перешел в альтернативные. **LXF**

Валентин Синицын » Главный редактор info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Web-сайт: www.linuxformat.ru

- **Адрес редакции:** Россия, Санкт-Петербург, Лиговский пр. 50 корп. 15
- **Телефон редакции:** (812) 640-49-90. Дополнительная информация на стр.128

LINUX FORMAT

Ноябрь 2008 № 11 (111)

Содержание

Весь номер – прямо как на ладони: приятного чтения!

Обзоры

KDE 4.1 12
В конце туннеля KDE наконец-то забрезжил свет. Будем надеяться, что это не поезд...

Ankh: Heart of Osiris 14
Отправляйтесь в древний Египет и поднимите свои логические способности на недосягаемую высоту.

› **Немногие догадываются, что древние египтяне частенько захаживали в коктейль-бары.**

Opera 9.5 15
Добро пожаловать в норвежский экспресс – Opera привносит в web-серфинг новые функции, не жертвуя ни граммом производительности.

Synology DS207+ 16
Создайте сетевое RAID-хранилище под управлением Linux, взяв за основу эту симпатичную коробочку.

Runtu 3 Desktop 17
Linux для тех, у кого нет широкополосного доступа в Интернет, но почему-то без проприетарных видеодрайверов.

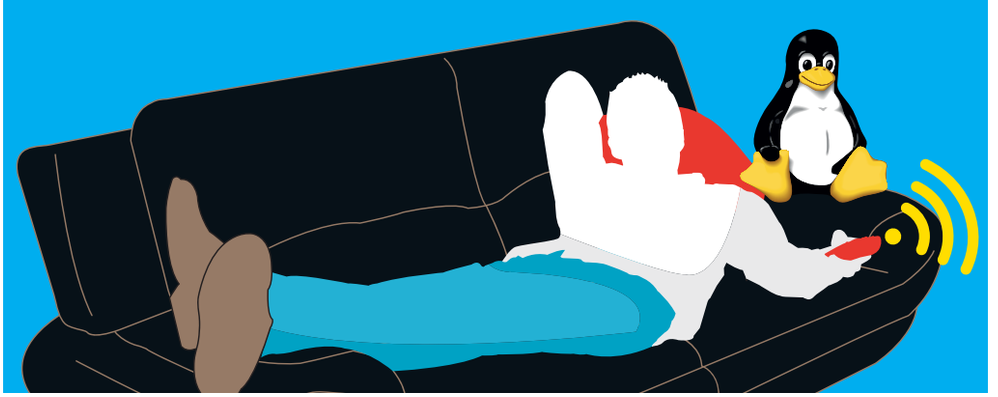
SlickEdit 2008 18
Это как удобная пара старых ботинок, только не для прогулок, а для программирования.

Сравнение: Как привести дела в порядок

Basket 24
Chandler 23
Emacs Org-mode 21
GTD-Free 21
Kontact 24
MonkeyGTD 22
ThinkingRock 23
Tracks 22

Наш ответ: MythTV

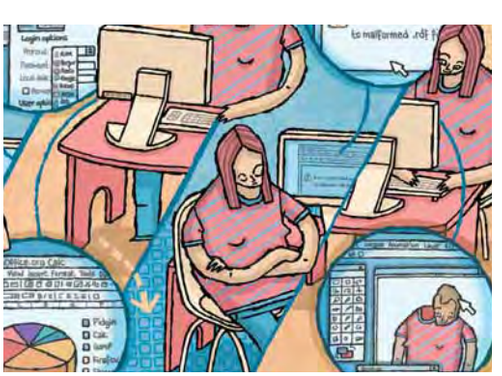
Откиньтесь на спинку кресла и позвольте Linux помочь вам расслабиться перед телевизором с. 32



Что за штука... DjVu ?

Новый формат изображений, предназначенный для архивных целей, разумеется с. 52

Проверка на девчонках с. 40



Люди говорят

> Мы решили, что работа должна доставлять удовольствие, и я надеюсь, нам удалось этого добиться

Карл Ворт о маленьких радостях векторной графики с. 28

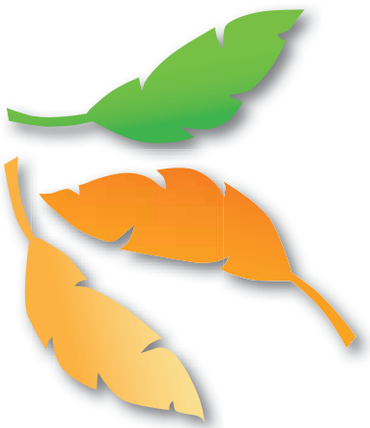
Спецрепортаж

30 дней в Haiku 48
Интригующий клон BeOS намерен показать, что Open Source – это не только Linux.

А ТАКЖЕ

Тест на подружке 40
Разработчики, прочтите внимательно: от этого зависит будущая популярность Linux!

Сборка из исходников 44
Не ждите, пока авторы вашего дистрибутива подготовят двоичные пакеты – сделайте это сами, как настоящий линуксоид.



СЭКОНОМЬ НА ПОДПИСКЕ!

Постоянные рубрики

Новости 04
События мира Linux глазами наших экспертов.

День свободы ПО 06
Вести со всех уголков России

Distrowatch 26
В этом месяце для вас: Debian, CCux 0.99 и Musix.

Интервью Linux Format 28
Карл Ворт, проект Cairo.

История успеха 30
ООО «Элсис», г. Нижний Новгород

Что за штука 52
DjVu: новый формат изображений, который сэкономит человечеству годы скачиваний и петабайты трафика.

Рубрика для админа 54
Д-р **Крис Браун** тестирует производительность файловых систем и наслаждается прелестями виртуализации.

Игрострой LXF 96
Осваиваем процедурные текстуры.

Школа LXF 100
Наша образовательная рубрика.

Ответы 106
Проблемы Linux решены: обновляем прошивку сканера, разбираемся с беспроводной сетью и нарезаем диск на разделы.

LXF HotPicks 112
Горячие новинки мира Open Source.

Диск Linux Format 118
Прочтите тему номера, поставьте Mythbuntu и садитесь к телевизору.

Через месяц 128
Узнайте, что мы заготовили на финальный номер 2008 года!

› **Приведите дела в порядок при помощи Linux, и проводите больше времени с друзьями в пивной.**

Учебники

Начинающим
GDesklets и двойная загрузка 62
Освежите бесполезную машину с Vista новеньким с иголочки Linux'ом, а затем добавьте к нему виджеты, как в OS X.

GIMP
Применяем выделения 66
Долой ножницы – пережиток прошлого века; режьте свои изображения в GIMP с куда большей гибкостью.

Мультимедиа
Советы по HDR 70
Погрузитесь в наложение тонов с *PFSTools*, будьте неконкретными в *DigiKam*, вещайте по сети с *Icecast* – в общем, встречайте новую серию учебников по мультимедиа!

Организованность
Синхронизируем календари 74
Держите важные данные под рукой – синхронизируйте настольный календарь с мобильником, чтобы не пропустить важную встречу.

Hardcore Linux
Nagios 78
Когда крупные сети ломаются, они ломаются по-крупному. Следите за сетевым трафиком и будьте на шаг впереди с Nagios.

CMake
Новые возможности 82
Нет, версия *CMake* осталась прежней, но сегодня мы обучим старого пса новым трюкам: в этом нам помогут расширения.

KDE 4
Локализация 85
Научим ваши программы говорить на любом языке без артефактов вроде несогласованных падежей и канцелярита.

R
GUI-интерфейсы 88
Десять графических помощников в нелегком деле статистического анализа данных.

gLite
Подключаемся к гриду 92
Узнайте, что кроется за модным ныне термином и получите представление о том, как все это использовать.

› **Наслаждаться чем-то навроде плазмоидов можно не только в KDE.**

LINUX FORMAT Новости

» Рубрику ведет Илья Шпаньков

ГЛАВНЫЕ НОВОСТИ: » Linux становится все мобильнее » ODF vs OO-XML: и вновь продолжается бой! » Планы на GNOME 3.0 » OpenOffice.org 3.0 и Mono 2.0 » Linux-репозиторий от Google

Linux-мобилизация продолжается

Одна из тем, которая активно обсуждалась на тематических ресурсах в этом месяце – использование GNU/Linux на мобильных устройствах. Самым заметным событием стало переименование фирмы Trolltech, приобретенной компанией Nokia, в Qt Software. Платформа для мобильных устройств Qtopia теперь называется Qt Extended, а новый официальный слоган Qt немного «подрос»: “Code less. Create more. Deploy everywhere” [Пишите меньше. Создавайте больше. Внедряйте везде]. Изменился также дизайн сайта бывшей Trolltech – оттуда убрали все упоминания старого брэнда [но доменного имени это не коснулось, – *прим. ред.*]. Нововведения затронули только названия и логотипы – условия распространения ПО не меняются, и нет оснований предполагать, что это произойдет в ближайшем будущем. Параллельно были подготовлены свежие релизы: *Qt 4.4.3* и *Qt Extended 4.4*. Первый из них содержит преимущественно исправления ошибок, а о втором необходимо сказать подробнее. Архитектура *Qt Extended 4.4* стала модульной, появились дополнительные возможности работы с сенсорными панелями и IP-каркас на базе *Telepathy*, а также инструментарий для автоматического тестирования системы на целевом устройстве – *Qt UI Test*. На наш взгляд, поводов для опасений нет, но некоторые пессимисты считают, что Nokia интересует только ПО для мобильных устройств и пора начинать опасаться за судьбу настольного *Qt*. Так это или нет, покажет время.

Вообще, Nokia «отметилась» на новостных порталах не только в связи с переименованием Trolltech. Директор компании по открытым проектам Ари Яаакси [Ari Jaaksi] заявил, что в следующем релизе Maemo (мобильная Linux-платформа для интернет-планшетов Nokia N8xx) появится поддержка технологии 3G/HSPA. Кроме того, реализацию ПО для работы в сетях третьего поколения финский гигант планирует передать в основное ядро Linux. Еще в Maemo 5 включат средства работы с цифровыми HD-камерами, некоторые проприетарные компоненты заменят на открытые и добавят профиль A2DP в Bluetooth-стек Bluez.

Следующий «виновник торжества» – поисковый гигант Google, а точнее, мобильная платформа Android. Итак:

» Вышел Android 1.0 SDK Release 1. В основном он содержит исправления ошибок, обнаруженных в версии 0.9.

» В блоге одного из разработчиков появилась информация о том, что Google планирует создать Android Market – систему распространения ПО для новой платформы.

» Публике представлен T-Mobile G1 (известный как HTC Dream) – первый в мире коммуникатор, построенный на платформе Android. По заявлению представителей компании, число предварительных заказов на устройство втрое превысило первоначальные прогнозы.

» Популярное приложение для GPS-навигации iGo находится в стадии портирования на Google Android.

В общем, по количеству упоминаний на новостных сайтах Android явно обогнал Nokia.

Еще одно событие, несомненно, заслуживающее упоминания – выпуск тестовой версии мобильного браузера от Mozilla Labs. Продукт назвали *Fennec*, а его интерфейс чем-то напоминает обозреватель для Apple iPhone. В настоящее время программа работает только на Nokia N8x0 (Maemo). О поддержке прочих мобильных Linux-платформ пока ничего не сообщается, а версия для Windows Mobile ожидается через несколько недель, возможно, уже к тому моменту, когда вы возьмете в руки данный номер журнала.

В заключение скажем несколько слов и о «третьем ките» мобильного Linux: проекте Openmoko и его коммуникаторе Neo FreeRunner. Во-первых, это устройство и аксессуары к нему (в частности, отладочную плату) теперь можно официально приобрести в России; продажами занимается компания Linuxcenter.Ru. Во-вторых, Давид-Рейес Самблас Мартинес [David-Reyes Samblas Martinez] анонсировал дистрибутив FDOM, предназначенный для работы на Neo FreeRunner. Давид-Рейес намеревается собрать в одном месте все приложения для Openmoko. В состав его детища входят: *Mozilla Minimo* и *Firefox*, *MPlayer* и различные мультимедиа-кодеки, *Pidgin*, *TangoGPS* и многое другое. В качестве оконного менеджера выбран *Enlightement*. Наконец, в Китае запущена учебная программа *OpenLab* по проектированию мобильных устройств, основанных на платформе Neo FreeRunner. В рамках курса студенты будут разрабатывать собственные проекты, используя Openmoko. Учебный план выпущен под лицензией CC-SA.

Как видите, нынешняя осень была очень «урожайной» на интересные события. Думаем, что и зима в этом смысле нас не подведет, поскольку производители сейчас заинтересовались мобильными устройствами с GNU/Linux на борту. Это не может не радовать.

Новости короткой строкой

» Google создал собственный репозиторий ПО для Linux: в настоящее время оттуда можно загрузить Picasa for Linux 2.7 и Google Desktop Linux 1.2. Поддерживаются следующие дистрибутивы: Ubuntu 7.04 (Feisty), Debian GNU/Linux 4.0, OpenSUSE 10.2 и Mandriva 2007 Spring.

» На конференции Linux Plumbers продемонстрирована Linux-система, полностью загружающаяся за 5 секунд.

» Вышла версия 2.6 популярного графического редактора *GIMP*. Основное нововведение: программа использует движок GEGL.

» Правительство Венесуэлы закупает миллион субноутбуков с предустановленным Ubuntu Linux.

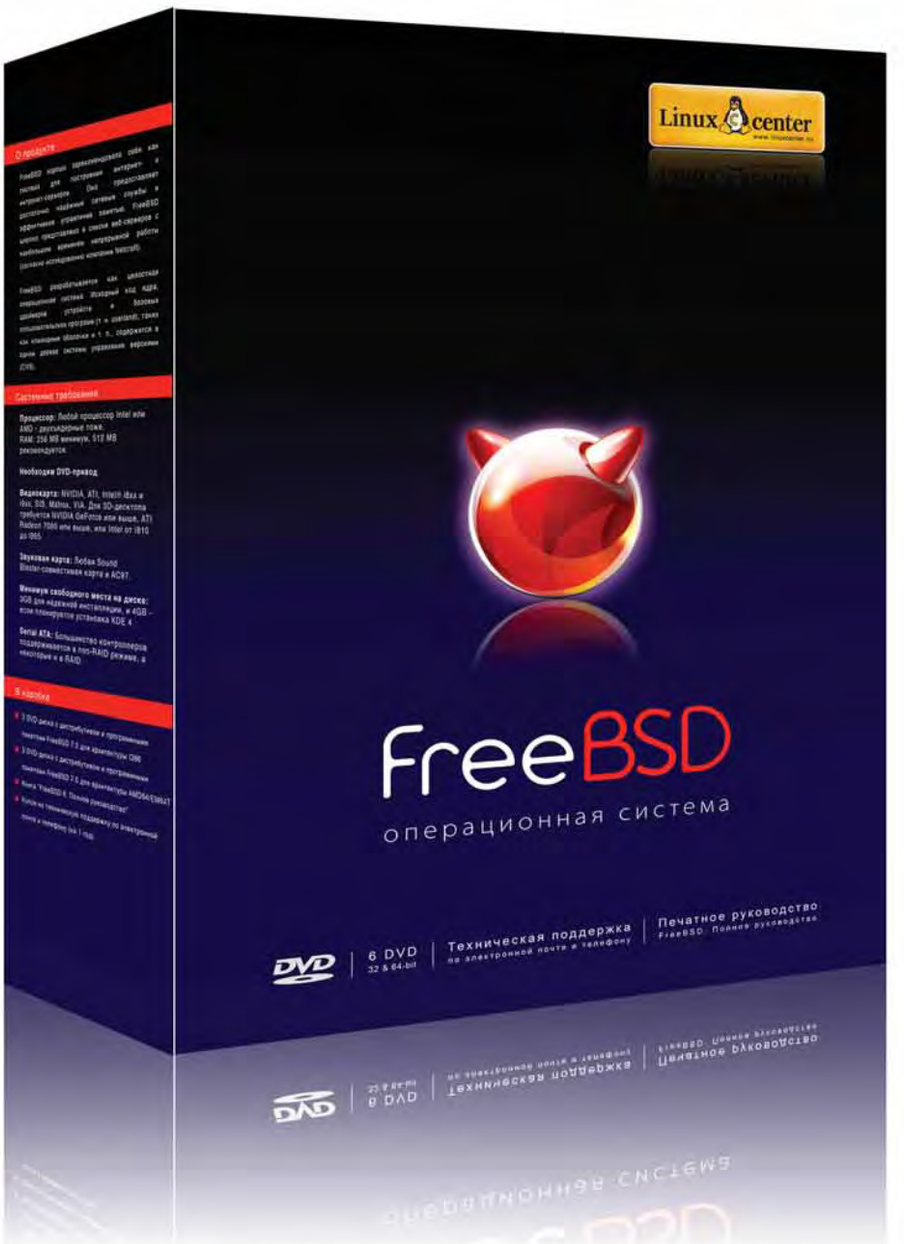
» Движок *Webkit* прошел тест Acid3. Другие популярные браузеры этим пока похвастаться не могут: у *Opera* 99 %, у *Firefox* и *Google Chrome* – 87 %. Меньше всех соответствует стандартам *Internet Explorer 8 Beta 2*, который набрал всего 21 %.

» Код Atheros HAL для чипсета 802.11a/b/g теперь доступен под GPL-совместимой лицензией ISC.

» Фьючерсная биржа CME Group вошла в состав Linux Foundation. Это самое крупное финансовое учреждение, состоящее в данной организации.

» В связи с нехваткой ресурсов и задержкой выпуска Gentoo 2008.0, релиз 2008.1 решено отменить.

» Увидели свет Mandriva Linux 2009, ALT Linux 4.1 Desktop и ядро Linux 2.6.27.



Надежность и эффективность

Операционная система FreeBSD 7.0
Печатное руководство
Техподдержка от ГНУ/Линуксцентра

Купите на сайте www.linuxcenter.ru или у партнеров!
Список магазинов партнеров: www.linuxcenter.ru/mag.phtml

Открытый офис: война на весь мир

Мы уже писали о том, что комитет ISO отклонил апелляции против принятия стандарта Office Open XML, предложенного компанией Microsoft (LXF110). После этого в Сети появилось множество сообщений об открытых (в той или иной степени) офисных форматах — ODF и OOXML. Предлагаем вашему вниманию дайджест наиболее интересных новостей и заявлений.

» По сообщению ODF Alliance, Шведский институт стандартов (SIS) утвердил Open Document Format в качестве национального стандарта. Напомним, что ранее ODF был принят в Бразилии, Хорватии, Италии, Южной Корее и Южной Африке. Тенденция хорошая; хотелось бы надеяться, что и Россия не останется от нее в стороне.

» 13 членов норвежского Экспертного комитета по стандартам (Standard Norge) опубликовали открытое письмо, в котором объявили о своей отставке. Standard Norge рассматривал вопрос о голосовании Норвегии по OOXML, и эксперты считают, что Комитет предпочёл поставить коммерческую выгоду выше профессиональных интересов. В своем письме они называют OOXML «неподготовленным, непригодным и недостойным» и сожалеют о том, что он был принят вопреки их мнению. Ранее представители Венесуэлы, Южной Африки, Эквадора, Бразилии, Кубы и Парагвая сообщили, что не уверены в нейтральности позиции ISO.

» В ответ на принятие формата OOXML комитетом ISO/IEC, компания IBM пересмотрит условия своего участия во всех международных организациях. Напомним, что «Голубой гигант» был категорически против ратификации открытого стандарта от Microsoft.

Две следующие новости не имеют прямого отношения к ODF, однако пройти мимо них мы не могли. В Министерстве юстиции Бельгии идет процесс миграции на SUSE Linux и *OpenOffice.org*. За текущий год было внедрено 2500 рабочих мест.

А тем временем увидел свет долгожданный *OpenOffice.org 3.0*! Приводить весь список изменений не будем, но отметим поддержку ODF 1.2, улучшенную совместимость с *MS Office* (в частности, импорт файлов *Microsoft Office 2007* и *Microsoft Office 2008* для Mac OS X) и более тесную интеграцию с рабочим столом. Электронные таблицы теперь поддерживают 1024 колонки вместо 256, а также реализована возможность совместной работы над документами. Еще в *OOo 3.0* добавили поддержку файлов *Microsoft Access 2007* и улучшили поддержку VBA-макросов. Не забыты и пользователи Mac OS X – *OpenOffice.org* для этой ОС больше не требует наличия *X11*. Ждите пакетов в своем любимом дистрибутиве [некоторые из них, в частности, Arch Linux, успели включить *OOo 3.0* за несколько дней до официального релиза, чем немало удивили пользователей, – *прим. ред.*]!

Последняя новость не очень радостная: предпринимаются попытки передать попечительство над стандартом ODF из OASIS в подкомитет ISO/IEC SC 34, значительная часть экспертов которого (восемь из девятнадцати) являются работниками Microsoft, а еще двое представляют подконтрольную американскому софтверному гиганту организацию Ecma TC45. Это значит, что компания может получить контроль над развитием единственного свободного конкурента OOXML.

Microsoft пытается продвигать собственный формат, чтобы не пользоваться общепринятым. Что ж – ничего нового в этом нет. Чем закончится очередная «война стандартов», пока сказать трудно, так что следите за новостями: мы обязательно будем освещать эту тему в следующих номерах **LXF**.

Третий GNOME уже не за горами

22 сентября, строго по расписанию, вышел очередной релиз популярного окружения рабочего стола GNOME. В версии 2.24 произошло немало интересных изменений: добавлены IM-клиент *Empathy* и SIP-телефон *Ekiga 3.0*, доработан *Deskbar*. В *Nautilus* появились вкладки, а также новый компактный режим обзора файлов. Список нововведений можно продолжить, однако этот релиз интересен тем, что с него началась активная подготовка к переходу на *Glib/GTK+* 3.0. Теперь от третьего GNOME нас отделяют всего два промежуточных релиза. В отличие от его главного конкурента, KDE 4, изменения в GNOME вносятся постепенно. Подготовительные работы по переходу на новые *Glib/GTK+* будут завершены только в GNOME 2.28, в который включат все технологии, необходимые для выпуска версии 3.0.

Разумеется, увеличение старшего номера версии предполагает множество изменений, в том числе – визуальных. Как будет выглядеть новый GNOME? Если верить предварительному плану разработки интерфейса, в нем появится плавный переход от экрана GDM к рабочему столу, а также плавное появление/исчезновение обоев и панелей. Разработчики решили отказаться от использования заставки [splash screen] и добавить новый режим панелей – «док» [dock]. Теперь любителям «маковского» интерфейса не придется устанавливать Avant Window Navigator. Будет сильно переработан движок виджетов, осуществлен переход на использование сжатого начертания шрифтов и ввод второго цвета для подсветки надписей. Вместо темы Tango по умолчанию будут использоваться значки Manga (лицензированные по LGPL), куда включат пиктограммы в высоком разрешении.

Сейчас идут разговоры о том, что GNOME все больше зависит от кроссплатформенной среды Mono – свободной реализации платформы Microsoft .NET. На наш взгляд, это не совсем верно – ключевые компоненты окружения рабочего стола к Mono отношения не имеют. Но верно то, что все больше программ для GNOME пишут с использованием свободного аналога .NET. Пользователей этих приложений наверняка порадует известие о том, что 6 октября состоялся финальный релиз Mono 2.0 с *Gtk# 2.12*, а выпуск следующей «минорной» версии планируется уже в ноябре. Разработчики обещают большой прирост производительности приложений, а кроме того, полноценную реализацию API 2.0: Windows.Forms, ADO.NET, ASP.NET, наряду с различными возможностями C# 3.0 (в первую очередь LINQ) и Visual Basic 8. В Mono, хотя и не без оговорок, теперь можно запускать *Paint.NET* (**www.getpaint.net**) – сравнительно крупный (150 000 строк исходного кода) растровый графический редактор для Microsoft Windows. Все это говорит о зрелости Mono и, смеем надеяться, положительно скажется на качестве ПО для GNOME. LXF

LXF Репортаж

Software Freedom Day'08

Минувший сентябрь оказался чрезвычайно богатым на различные мероприятия, посвященные свободному ПО — команда LXF попыталась охватить их все в одном материале.

Для одних свободное ПО – идеология, для других – бизнес-модель. Эта разница во взглядах породила немало жарких споров на тематических ресурсах, но одна точка соприкосновения у противоборствующих сторон все же есть: свободное ПО – это движение. И как всякое уважающее себя движение, оно имеет ряд обязательных атрибутов: сочувствующих (сообщество), лидеров и, наконец, «профессиональные праздники», самым важным из которых, наверное, является День Свободы ПО – Software Freedom Day (www.softwarefreedomday.org). Это международное событие, отмечаемое ежегодно в каждую третью субботу сентября, празднуется с размахом: если в 2004 году в нем приняло участие 70 команд, то в этом – более 500. Приятно отметить, что 17 из них приходится на нашу страну. Опубликовать репортажи из всех этих городов было бы нереально, но мы с удовольствием предлагаем вашему вниманию подборку впечатлений участников SFD со всех уголков России! Помимо Дня Свободы ПО, конец сентября 2008 года ознаменовался большим числом фестивалей инсталляций и других неформальных встреч сторонников Linux. Мы решили, что это совпадение не случайно – ищите все подробности на следующих двух страницах.

Но – довольно предисловий. Вот что могут рассказать непосредственные участники событий:

» Анастасия Ильичева, соорганизатор SFD'08 – Саратов, 20 сентября 2008 года

Под крышей Саратовского государственного социально-экономического университета (СГСЭУ), в лучшей мультимедийной аудитории Поволжья, собралось рекордное число любителей и профессионалов свободного ПО. Ограничений на участие в празднике не было – организаторы, супруги Евгений Федин и Анастасия Ильичева, были рады каждому гостю. Самые юные участники фестиваля едва пошли в школу, а самые маститые помнят еще ламповые компьютеры.

› День Свободы ПО в Саратове прошел как веселый праздник, но в форме конференции.

Мы пригласили на праздник российских аутсорсеров, информационные издания, пропагандирующие свободное ПО. Почетным гостем саратовского фестиваля стал Александр Михайлович Поносов. Его визит был приурочен к одному из наиболее ярких и значимых событий – 19 сентября мэтр установил комплект СПО в гимназии №1 г. Саратова. Это учреждение стало первой ласточкой в области внедрения СПО в школы Саратовской губернии. На фестиваль пожаловала и «живая легенда *Amarok*» Вадим Петрунин (Фархип.Ру). Кстати, сотрудники этой компании показали, что они не только классно программируют, но и отлично разбираются в экономических и социальных аспектах свободного ПО – доклад Михаила Дьяконова на эту тему был удостоен приза зрительских симпатий.

Праздник прошел в формате конференции со стендовыми и секционными докладами. Каждый выступающий старался затронуть наиболее интересную, на его взгляд, тему, будь то разработка ПО, миграция на Linux целых компаний или обучение студентов. Живое обсуждение вызвали не только практические проблемы, например, поиск свободных ERP-решений, но и сугубо научные – грид-технологии и метод Монте-Карло.

Особый интерес аудитории вызвало подведение итогов Недели разработки СПО в СГСЭУ. Оригинальная инициатива старшего преподавателя Евгения Федина была с радостью подхвачена его студентами. Всего за одну неделю ребята сумели и музыку к фестивалю сочинить, и свободные аналоги логических игр написать, и прототип сайта подготовить, и даже свой дистрибутив собрать! По итогам Недели разработки, лучшие студенты получили предложения от работодателей прямо на фестивале. А те, кто только делает первые шаги на этом поприще, узнали о возможностях обучения на факультете информатики и информационных технологий в СГСЭУ, в «Академии АйТи» и, конечно же, общения и обмена опытом, например, в сообществе СарЛУГ.

По окончании конференции все гости получили диски со свободным ПО для Windows, а самые смелые – дистрибутивы Ubuntu 8.04.1 LTS Desktop Edition и Mandriva One 2008. Наиболее активные участники были награждены значками и наклейками с символикой фестиваля.

» Алексей Радионов, соорганизатор SFD'08 – Цимлянск, 20.09.2008

Организаторы нашего SFD – из города Волгодонска, но в качестве «площадки» для мероприятия была выбрана «МОУ СОШ № 2» города Цимлянска – победитель приоритетного Национального проекта «Образование» в 2007 году. Выражаем отдельную благодарность за инициативу Виктору Кривошлыкову, преподавателю информатики этой школы. На открытом уроке учащимся одиннадцатых классов было рассказано о положении дел в компьютерной отрасли в России и в мире, о законе и лицензиях, продемонстрирована операционная система Linux, а также прикладные свободные программы, работающие как в Linux, так и в Windows.

Честно говоря, мы не ожидали такого живого интереса к Linux со стороны школьников. Многие из них даже не подозревали о существовании свободного программного обеспечения, но после нашей лекции выразили готовность изучать Linux с целью его дальнейшего применения как в учебном процессе, так и в качестве ОС на домашнем компьютере.

Цимлянские школьники отнеслись к Linux с большим интересом.

«Гвоздем программы» стал рассказ о возможностях рабочего стола Linux с демонстрацией мультимедийных приложений для воспроизведения видео и музыки. Не были забыты и игры: ребята увидели культовую 3D-гонку *TuxRacer*, 3D-стрелялку *Nexuiz*, футуристическую аркаду *GridWars*. Мы донесли до школьников мысль, что принять участие в создании таких замечательных вещей и воспользоваться ими может каждый. Лучшим подтверждением стал доклад соорганизатора SFD Сергея Степанова, посвященный *Inkscape*, *GIMP*, *Blender* и программам, необходимым в учебе и дома – пакете *OpenOffice.org* и браузере *Firefox*.

Интерес учащихся говорит о том, что информация о свободном ПО не всегда достигает конечного пользователя. Именно поэтому такие встречи необходимо сделать регулярными – такого мнения придерживается и третий организатор встречи, сисадмин со стажем Александр Пащенко. Мы с удовольствием поделимся презентационным материалом для желающих устроить подобный урок открытого ПО в своей школе или вузе. Пишите нам по адресу: pretorean@gmail.com.

Никита Лялин, соорганизатор SFD'08 – Екатеринбург, 27.09.2008

День Свободы ПО в этом году проходил в Екатеринбурге впервые. Семинар, проводимый в здании Уральского государственного университета (УрГУ), планировалось разделить на две части, с перерывом на небольшой обед. Первая часть была посвящена общей тематике, вступительное слово было дано Илье Шпанькову (Opera Software), озвучившему четыре основные свободы, присущие свободному ПО. Это не заняло много времени, в отличие от следующей презентации Александра Трубина (Sun Microsystems), который изложил историю появления и развития течения FOSS более подробно. Он также рассказал о текущем положении дел в этой сфере, поделился своими размышлениями о будущем. Последняя часть выступления поведала посетителям о вкладе, который вносит компания Sun Microsystems в свободное и открытое ПО. После этого слово было вновь передано Илье Шпанькову, который в форме диалога с аудиторией, рассказал о коммерческой стороне применения свободных и открытых продуктов. Немалое внимание в его докладе было уделено преимуществам использования FOSS в образовании.

Те, кто захотел остаться на вторую часть семинара, перешли в другую аудиторию, и мероприятие продолжилось. Желающие получили промоматериалы: диски с *OpenOffice.org 2.4.1-1 Pro*, Mandriva 2008.1, NetBeans и OpenSolaris и журналы Linux Format. Первый прозвучавший доклад был посвящен настольному применению СПО. Сначала Степан Кузьмин (Екатеринбургский LUG) продемонстрировал одну из свободных игр, которая входит в состав LiveDVD-дистрибутива linuX-gamers.net, затем выступил ваш покорный слуга с докладом о KDE 4.1. Следующий доклад сделал Игорь Стадухин (Уральская академия информационных технологий) – его презентация была посвящена применению дистрибутива Smoothwall Express в сетях малых предприятий. После этого эстафету переняла Елена Ушакова (Ru.OpenOffice.org, MyOOo.ru): она поделилась собственным опытом перевода предприятия с *Microsoft Office* на *OOo*, обозначила главные шаги миграции, возникающие при этом проблемы, дала несколько рекомендаций о том, как уладить юридические и налоговые вопросы.

Интерес к СПО в Екатеринбурге оказался выше, чем ожидали организаторы: в зале яблоку было негде упасть.

Алексей Сиваков, организатор Mandriva & openSUSE Install Party — Краснодар, 28.09.2008

Мероприятие было организовано и проведено силами участников Краснодарского подразделения Северо-Кавказской группы пользователей Linux (NCLUG), на помощь которым приехали их коллеги и единомышленники из Пятигорска и Ростова-на-Дону. Место для проведения встречи было любезно предоставлено компьютерным салоном «Computer-City»; кстати, эта фирма использует Linux для демонстрационных и некоторых внутриофисных задач. А компания Mandriva.Ru выслала нам партию фирменных заводских дисков с Mandriva One для бесплатной раздачи всем желающим.

Фестиваль начался с торжественного вывешивания баннеров сообщества NCLUG и Mandriva (в этом деле приняли участие почти все присутствующие) и краткого вступительного слова одного из координаторов NCLUG Алексея Сивакова aka Goodvin. После этого закипела работа. Было роздано около сотни дисков с Mandriva One, записано несколько десятков копий Mandriva Free и OpenSUSE, а также различных обновлений и дополнительных пакетов. Несколько дистрибутивов установили прямо на принесенные участниками ноутбуки. Отдельно хочется отметить большой интерес, проявленный посетителями к специализированному игровому LiveDVD от команды linuX-gamers.net.

По итогам мероприятия все получили массу удовольствия и набрались интересного опыта; были заведены новые знакомства, пришло понимание востребованности, необходимости подобного рода акций и желание проводить их на регулярной основе: так, Rostov-Don GNU/Linux Demo-Party запланирован уже на 12 октября.

Как видно, сентябрь прошел не зря. Но все же 17 из 500 – это только начало. Ждем не дождемся 2009 года! LXF

Групповое фото на память под собственноручно вывешенным баннером – особое удовольствие.

LXF Репортаж

Hackontest: как это было...

Разработчик векторного графического редактора *sK1* **Игорь Новиков** делится своими впечатлениями от соревнования разработчиков свободного ПО, проведенного под патронажем Google.

Соревнованиями и конкурсами, связанными с Open Source-технологиями, сегодня вряд ли кого-то удивишь. Тем не менее, первая редакция Hackontest, состоявшаяся в этом году, привлекла к себе внимание.

Необычность этого соревнования состоит прежде всего в том, что оно организовано под патронажем «небольшой» компании по имени Google Inc., которая уже давно славится своим неравнодушным отношением к свободному ПО. Вторая немаловажная деталь – конкурс проходил в Швейцарии, а кто же откажется от такой возможности посетить легендарную страну банков, шоколада и одноименного сыра?

Формула состязания такова: три команды, прошедшие отборочный тур, прибывают в Цюрих, где на выставке OpenExpo за 24 часа выполняют задание жюри, находясь при этом в закрытом контейнере eToy.TANK. Победителем считается та команда, которая (по мнению жюри) покажет наиболее значимые результаты.

Первый отборочный этап Hackontest'а происходил в Интернете (http://hackontest.org) – пользователи и разработчики различных открытых проектов регистрировали свои приложения, предлагали задания и голосовали за их важность с точки зрения дальнейшего развития проектов. Поначалу голосование шло вяло. Наибольшую популярность снискали представители мира свободной графики: *Inkscape*, *Scribus*, *GIMP*, *sK1*. Но, поскольку пользователи особо не стеснялись в своих пожеланиях и запросили, по мнению разработчиков, достаточно серьезные изменения, которые нереально воплотить за 24 часа, активность со стороны многих проектов упала до нуля.

В июле (последнем месяце голосования) картина резко изменилась. На вершину рейтинга поднялись web-приложения: *phpMyAdmin*, *TYPO3*. Из графических программ остался только наш проект *sK1*, и началась довольно агрессивная конкуренция. Был случай, когда пользователи *TYPO3*, обнаружив возможность голосовать «в минус», начали понижать рейтинг *sK1*. Наши сторонники в долгу не остались, вогнав в еще больший минус *TYPO3* и, заодно, *phpMyAdmin*. В итоге, организаторы запретили отрицательные баллы и нейтрализовали уже проставленные «минусы». Тем не менее, *phpMyAdmin* сошел с дистанции, и его место занял проект *OpenLieroX* (игра в стиле *Worms*).

› **Фото на память: участники и организаторы конкурса Hackontest.**

В ночь на 1 августа (конечный срок голосования) проект sK1 шел впереди с хорошим отрывом от TYPO3. Тройку лидеров, с большим отставанием, замыкал *OpenLieroX*. Именно так первый этап и закончился: на первом месте – *sK1*, на втором – *TYPO3*, на третьем – *OpenLieroX*.

Август и начало сентября прошли в бумажной волоките: для получения виз мне, как руководителю команды, пришлось собрать уйму документов (когда мы их посчитали, получилось около восьмидесяти различных «бумажек»). И швейцарское посольство оказалось довольно бюрократичным – пришлось дважды ездить в Киев на сдачу документов. Для сравнения, визу во Францию, на конкурс Trophées du libre мне выдали спустя 15 минут с момента подачи всех необходимых бумаг. Тем не менее, все члены команды *sK1* успешно получили визы, и 23 сентября мы вылетели в Цюрих.

Встречают по одежке

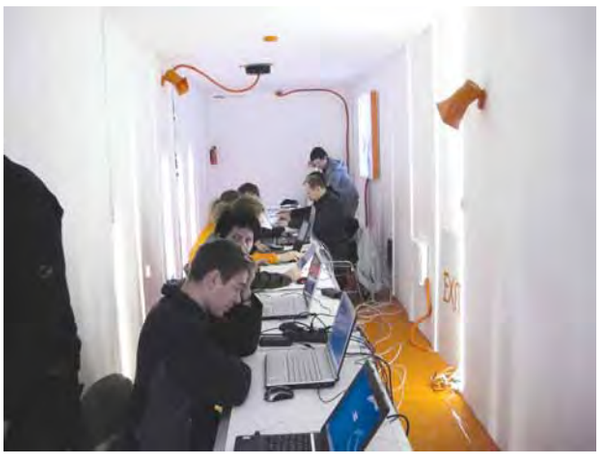
Уже при подготовке к поездке в Швейцарию стало ясно, что организация конкурса, несмотря на патронаж Google, довольно-таки сумбурна. Это, видимо, следствие того, что соревнование проводилось в первый раз. Организационно Trophées du libre на порядок выше: начиная с подготовки необходимых документов, приезда участников и заканчивая распорядком в процессе самого мероприятия.

В Цюрих мы приехали поздно вечером, но в аэропорту нас не встречали. Нам нужно было самостоятельно добраться до небольшого городка Винтертур, который находится недалеко от Цюриха (20 минут езды на поезде). Задача эта несложная, но учитывая, что никто из нас никогда до этого не был в Швейцарии и мы были совершенно не в курсе, что и как, такой «квест» добавил адреналина. Во Франции данный вопрос был проработан более основательно – все участники (как жюри, так и конкурсанты) собирались в аэропорту и организованно ехали на специально нанятом автобусе. Это, конечно, снимало многие проблемы.

В Винтертур мы добрались (с небольшим опозданием), сверяясь при этом с GPS. Поезда в Швейцарии ходят очень быстро, на станциях стоят недолго, так что мы боялись пропустить нашу остановку. Но, к счастью, наша цель – достаточно крупная станция, и высадились мы вполне успешно. Выйдя в город, а в это время уже было около девять вечера, мы обнаружили, что в районе, где должна располагаться выставка OpenExpo, на улицах практически нет людей. Сделав круг вокруг экспоцентра, мы вышли все-таки к контейнеру Hackontest.

Сказать по правде, моим первым ощущением было разочарование. Это был обычный контейнер, внутри которого рядком были обычные столы и стулья. Питание для ноутбуков – через удлинители, со всех сторон висят провода локальной сети. Собственно, это и весь антураж конкурса. Вход в контейнер был закрыт плексигласом, и попасть вовнутрь почему-то можно было только на корточках через круглое отверстие. Зачем так придумали – остается загадкой. В общем, разительный контраст со средневековым поместьем в Суассоне, где проходил конкурс Trophées du libre. Настроив доступ к Интернету и обеспечив себе возможность демонстрировать экраны наших ноутбуков на большом мониторе в помещении выставки, мы отправились в гостиницу.

› Это тот самый холодный контейнер, где все и происходило.



› Работа над новым кодом в самом разгаре.

Кодинг-марафон

На следующее утро началось само соревнование. Все конкурсанты собрались в контейнере, подключили компьютеры и стали ожидать задания жюри. Первый казус – в контейнере оказалось жутко холодно. Климат в Швейцарии довольно суровый (через неделю после конкурса уже выпал снег) и в контейнере с утра было довольно зябко, даже в верхней одежде. Второй казус – задание от жюри так и не пришло. Точнее, организаторы сообщили, что мы сами можем выбирать себе задание, желательно согласно пользовательскому рейтингу. К чести организаторов, через какое-то время принесли тепловентиляторы, и в контейнере стало значительно теплее. Хотя, должен отметить, программировать в перчатках мне еще не приходилось... Конечно, организаторы всячески пытались сгладить неудобства и поддержать конкурсантов – пицца, кофе, кола, тонизирующие напитки, обед прямо в контейнере над клавиатурами ноутбуков – все в лучших традициях историй про хакеров.

Мы выбрали достаточно амбициозное задание: «Поддержка CMYK-изображений в документе», а в довесок к нему второе – «Поиск и замена текста с использованием регулярных выражений». Поддержка CMYK – достаточно популярная тема; аналогичное предложение было сделано и команде *GIMP*, но разработчики ответили на него категорическим отказом, мотивировав это тем, что в 24 часа с такой работой не уложиться.

В случае *sK1* данную задачу можно было решить, поскольку у нас растровые изображения модифицируются не в отдельных частях, а все целиком. Да и базовый язык разработки – не C, а высокоуровневый Python, что сильно упрощает реализацию: воплощение сложного алгоритма на Python может получиться десятикратно более компактным, чем на C++/Java. Отталкиваясь от этого, мы и приступили к делу.

В итоге, к вечеру у нас уже была работающая поддержка CMYK, RGB, RGBA, Grayscale и монохромных изображений. Мы не ограничились только CMYK, поскольку остальные варианты цветовых пространств связаны с ним и переделывать реализацию после конкурса не было никакого желания. Поддержку цветовых пространств сделали «сквозной» – не только на экране, но и в формате **sk1** и в выводе на печать через PDF. Сохранение и чтение из документов **sk1** доделывали уже глубокой ночью: пришлось изменить формат записываемого в файлы ***.sk1** изображения, поскольку текущий вариант базировался на JPEG-формате, что не охватывало все необходимые варианты и, плюс к этому, добавляло сжатие с потерями.

Также к вечеру был готов модуль для поиска и замены текста. Иными словами, к утру мы уже могли похвастаться двумя выполненными заданиями, одно из которых было весьма существенным (по условиям конкурса предусматривалось всего одно). Но для полной завершенности тематики поддержки CMYK-изображений не хватало «маленькой детали», а именно, управления цветом для растровых объектов. Сама по себе тематика управления цветом достаточно сложна и, например, *Inkscape* обзавелся ее частичной поддержкой лишь прошлой весной.

В нашем приложении векторная графика уже давно отображается с использованием библиотеки *LittleCMS*. С растровыми объектами ситуация сложнее, поскольку для корректного управления цветом необходимо пересчитывать каждый пиксель исходного изображения. Написать и отладить родной модуль для данной операции за ночь, после 12 часов программирования – задача нереальная, но тут нам повезло. В поиске обнаружился модуль *pyCMS*, созданный в 2002 году и давно заброшенный автором. Естественно, он не компилировался, поскольку код сильно устарел. Более того, автор писал его под Microsoft Windows и даже не пробовал под другими платформами.

Я решил рискнуть (благо свободные часы до 10 утра позволяли) и попытался отладить код модуля. И риск оправдался – после довольно небольшого числа изменений модуль начал компилироваться, более того – практически с первой попытки заработало управление цветом; такого результата я даже не ожидал! Это был серьезный успех, так как изменения, внесенные в код *sK1*, вполне тянули на причину для релиза.

Подведем итоги

В 10 утра нас попросили завершить кодирование, выложить результаты на сайты своих проектов и... готовить презентацию о проделанной работе. По большому счету, это было неправильно: после 24-х часов программирования, состояние участников было совсем «никакое». Естественно, кто был помоложе, тот и чувствовал себя получше, но ко мне это не относилось. Хотелось только одного: побыстрее добраться до гостиницы и выспаться. В итоге, презентация получилась смазанная. К тому же, организаторы установили проектор для ее проведения в контейнере и при дневном освещении было практически нереально продемонстрировать управление цветом и отличие разных типов растровых объектов.

Итоговая оценка жюри вообще поразила: внимание обращалось на количество написанных строчек кода и активность в ходе презентации. Тут мы проигрывали во всем статьям. Код на Python, как упоминалось ранее, крайне компактный. А по внешнему виду на хакеров мы явно не тянули: ни тебе рваных джинсов, ни бороды, ни колец в носу... Вот и вышло нам почетное третье место, а «золото» и «серебро» досталось *OpenLieroX* и *TYPO3*, соответственно.

Тем не менее, разочарования нет: код, созданный во время конкурса – достаточно качественный. Буквально на следующий день после соревнования все изменения были перенесены в основную ветку проекта. Результат дает основания полагать, что проект теперь имеет полноценную систему управления цветом – практически такую же, как в профессиональных приложениях для полиграфии. А спустя буквально неделю после состязания был улучшен импортер формата CDR: на базе расширенного API получился корректный импорт растровых изображений из формата CorelDRAW. Поэтому, с точки зрения развития проекта, результаты Hackontest'а можно назвать отличными.

Ну и конечно же, не стоит сбрасывать со счета само посещение Швейцарии. После экскурсий в Цюрих, Люцерн, поездки в Альпы от негативных впечатлений не осталось и следа. В общем, несмотря на то, что мы не «настоящие хакеры», результат, по нашему мнению, отличный, а третье место... ну, так уж получилось. LXF

LINUX FORMAT

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов

Алексей Федорчук
Его слабости – mass storage, разметка диска и файловые системы.

Настольный FreeBSD

Если «десктопизацию» Linux можно считать свершившимся фактом, то в отношении FreeBSD картина вырисовывается более противоречивая. С одной стороны, тенденция в сторону рабочего стола не миновала и эту ОС – как саму, так и ее спецответвления вроде PC-BSD и DesktopBSD. С другой же, все громче слышен грозный ропот админов: «FreeBSD – это святое, юзеры, уберите от нее свои грязные десктопные лапы». Где же истина?

Как обычно, посередине. Начнем с того, что админы совершенно необоснованно присвоили себе эксклюзивные права на FreeBSD, в чем легко убедиться, вспомнив историю. Ведь это была первая из берклианских систем, ориентированная на самую демократическую платформу того времени – i386, каковая, очевидно, и предназначалась для столь же демократической публики, то есть для пользователей. Правда, тогда довольно специфических, все больше специалистов в области Computer Science, но тем не менее, никаким боком не системных администраторов. То, что последним удалось прикрутить эту ОС к своим потребностям, лишь идет в плюс и ей, и им, но не дает ни малейших прав собственности.

Ну а создатели настольной FreeBSD просто пошли проторенным путем создателей «дружелюбных» дистрибутивов Linux, без учета специфики FreeBSD (и специфики пользователей, для которых она создавалась).

Так не настал ли исторический момент вернуть FreeBSD на рабочие столы такой, какая она есть? Со всеми ее нынешними усовершенствованиями, но без новомодных вытребенок. Ведь (позволю процитировать самого себя) главная причина неиспользования FreeBSD на рабочем столе – в том, что ее мало кто там использует. Попробуем?

alv@posix.ru

Сегодня мы рассматриваем...

12 KDE 4.1
Команда KDE наконец-то выпустила столь необходимое обновление. Хватит ли его, чтобы остановить недовольных и оградить разработчиков от очередного полугода стонов и жалоб?

14 Ankh: Heart of Osiris
На бумаге игра выглядит просто прекрасно. Великолепная графика, управление в духе «найди и щелкни» – ну прямо *Monkey Island*. К сожалению, только на бумаге...

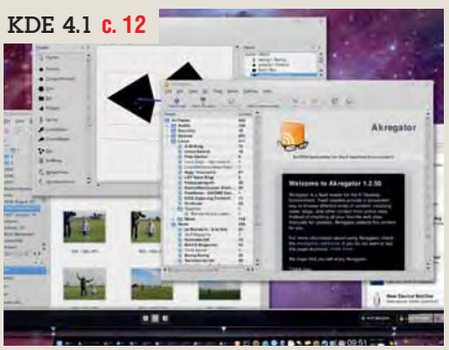
15 Opera 9.5
В дни вездесущего *Firefox*, бесплатный проприетарный браузер кажется сущим безумием, но *Opera* не устает доказывать, что слухи о ее смерти сильно преувеличены.

16 Synology DS207+
И снова сетевое хранилище данных, работающее под управлением Linux. Но у этого нет даже Linux-версии программы-установщика!

17 Runtu 3 Desktop
Нужно ли миру еще одно ответвление от Ubuntu? Кто знает... По крайней мере, проблема коричневого фона здесь, наконец, решена!

18 SlickEdit 2008
Шикарный текстовый редактор, созданный для программистов, которые знают, чего хотят, и к тому же зарабатывают достаточно, чтобы это себе позволить.

KDE 4.1 с. 12



› Станет ли релиз 4.1 возвратом KDE к былой славе или очередным разочарованием для верных пользователей?

Ankh: Heart of Osiris с. 14

› Сломайте себе голову ~~о стену~~ об эту зубодробительную смесь египетско-германского с калифорнийским.

НАШ ВЕРДИКТ: пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибальной шкале (10 – высшая оценка, 0 – низшая). Как правило, мы оцениваем функциональность, производительность, простоту использования и цену, а для бесплатных программ учитывается документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.

Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но, если разработчики рекомендуют Autopackage, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth

Разработчик: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Документация	9/10

» Если весь мир – сцена, то Google Earth – театр. Простая в использовании, захватывающая и ободряюще практичная программа.

Рейтинг 9/10

KDE 4.1

Версия 4.0 была чересчур незрелой по шкале **Ника Вейча**. Сможет ли 4.1 спасти лицо Kool Desktop Environment?

LINUX FORMAT На DVD
» Исходный код KDE 4.1

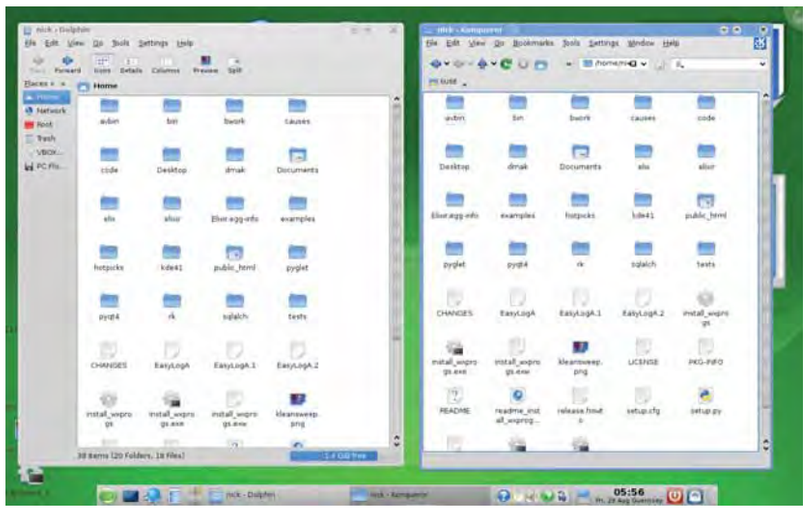
Вкратце...

» Графическое рабочее окружение; славится тем, что позволяет настроить все или почти все. См. также: Gnome, *Xfce*.

KDE 4.0, вышедший в январе нынешнего года, попал и в OpenSUSE 11, и в Fedora 9, и даже украсил однажды обложку *Linux Format*, но люди знающие прекрасно понимали, что это всего лишь опубликованная бета. И не удивительно: при таком количестве изменений процесс притирки неизбежен.

Некоторым притирка показалась слишком болезненной, и несмотря на пышность новой графики, налицо была нехватка функций и прикладных программ. Версия 4.1 – попытка устранить эти проблемы и дать KDE 4 шанс обрести популярность. Команда KDE не жалела усилий: между выходами версий зафиксировано более 20000 изменений в коде, что – надеемся – говорит о множестве устраненных ошибок и добавленных функций.

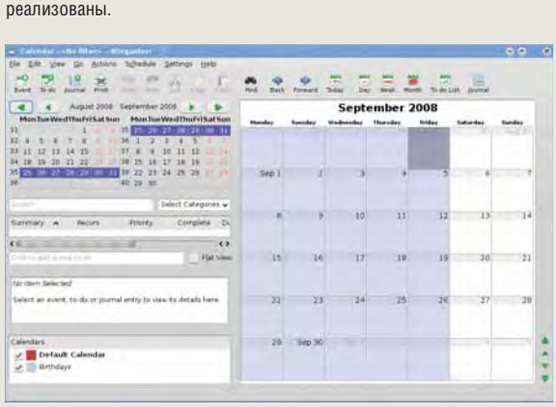
Не все, однако, получилось. Честно говоря, рабочий стол недоделан. Чего вы ждете, наводя указатель мыши на значок рабочего стола? Всплывающей подсказки или мета-информации? Может быть, миниатюрного изображения содержимого? А получаете панель меню плазмоидов для настройки и некоторых других операций, которая просто путается под ногами – да и то если виджеты разблокированы.



› **Новый *Dolphin*** (слева) **или старый добрый *Konqueror*? Запутаться нетрудно, ведь оба очень похожи... что будем брать?**

Ура, заработало!

Одна из основных жалоб про KDE 4.0 – на нехватку работающих приложений. Перевод прикладных программ с 3.x на 4.x – очень непростая задача, особенно в том случае, если разработчики хотят в полной мере воспользоваться новыми функциями. Для многих главная потеря – отсутствие в KDE 4.0 на момент релиза готового PIM-ПО. Теперь это в прошлом: *KOrganizer* и *KMail* получили новый формат, совместимый с KDE 4, хотя и не все первоначально заявленные функции пока реализованы.



› **Теперь даже KDE-пользователь не пропустит важной даты!**

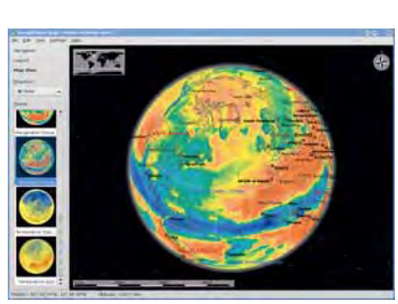
Неласковый Рабочий стол

Еще более раздражает разнобой в размерах значков на рабочем столе. Выход – использование нового виджета для отображения каталога **Desktop**, отчасти безумная концепция. Вместо привычной и ставшей классической метафоры рабочего стола, Folder View отводит для значков определенную зону экрана, а для работы – определенный каталог файловой системы. Двойной щелчок на каталоге открывает *Dolphin* для навигации, и вы получаете еще одно окно на рабочем столе. Попробуйте передвинуть его за строку заголовка – и вы поймете, что оно не только прозрачно, но и бесплотно. Попробуйте открыть Zip-файл или перетащить файлы по другому «KIO-адресу» – не выйдет, а значит, разработчикам KDE есть над чем трудиться.

> «Очень даже приятно иметь иконки разных размеров на рабочем столе.»

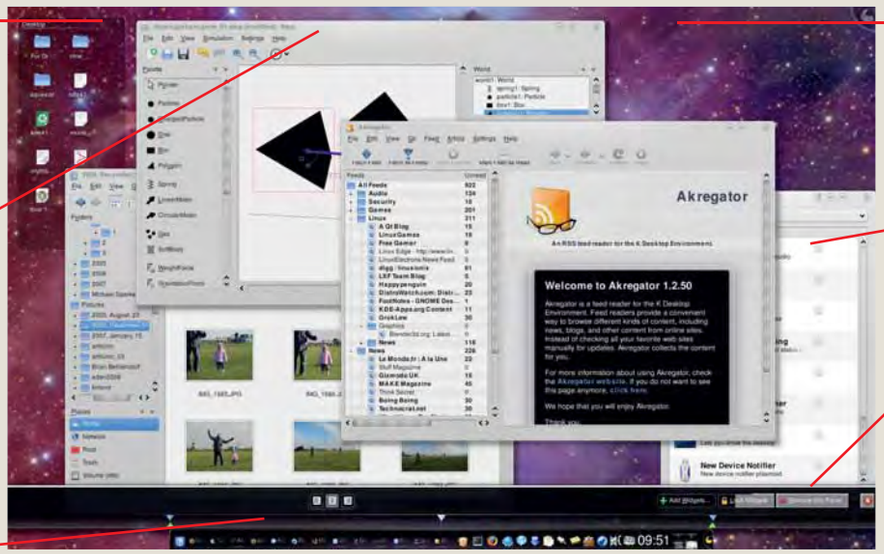
Есть тут и положительная сторона: иметь на рабочем столе значки разного размера иногда даже полезно. Новое общее свойство для плазмоидов и значков – масштабирование, и те, что поважнее, можно сделать побольше, а те, что используются реже – помельче. Масштабирование работает безупречно, даже при всем этом безумии. Но так как держать файлы на рабочем столе теперь невозможно, толку от этого новшества мало.

Другая важная часть рабочего стола – панель. Новый плазмоид-менеджер панели позволяет перемещать апплеты, изменять размеры панели и менять расположение значков, что, наконец, устранило проблемы предыдущей версии. Новое стартовое меню переработано для 4.0 (теперь оно называется “Kickoff”) и больше не менялось – в нем по-прежнему трудно что-либо отыскать, даже зная примерно, где оно находится. Можно переключиться к меню старого стиля, но за это придется заплатить: на данный момент единственный способ добавить любой «не-виджет» на панель – щелкнуть на нем правой кнопкой мыши, что в меню старого стиля недоступно. Парадокс: KDE часто



› **Заблудились? *Marble*, замечательный свободный атлас, поддерживает OpenStreetMap, так что до ближайшего газетного киоска доберетесь.**

Новое в KDE 4.1



«Каталожное» представление
Этот плазмоид позволяет выбрать любой каталог и отобразить содержимое на рабочем столе.

Новые приложения
Версия 4.1 получила несколько новых приложений, среди которых инструмент физического моделирования *Step*.

Обновленный Dolphin
Dolphin получил собственное иерархическое представление файловой системы.

Аниматор
Как и в Apple Dock, работающие приложения отображаются в анимированной панели, безупречно управляемой и масштабируемой.

Настройка панели
Теперь панель позволяет перемещать апплеты, масштабировать их и изменять расположение.

Стабильная плазма
Есть несколько новых плазмоидов, а можно импортировать виджеты Apple Dashboard.

Без значков
Рабочий стол больше не будет виртуальной папкой. Теперь можно только добавлять плазмоиды (по крайней мере, на столе будет чисто).

критикуют за обилие опций, в то время как адепты KDE жалуются на потерю контроля. Панель пока «не умеет» менять фон и автоматически скрываться, но мы надеемся, что это скоро будет исправлено.

Хлеб и зрелища

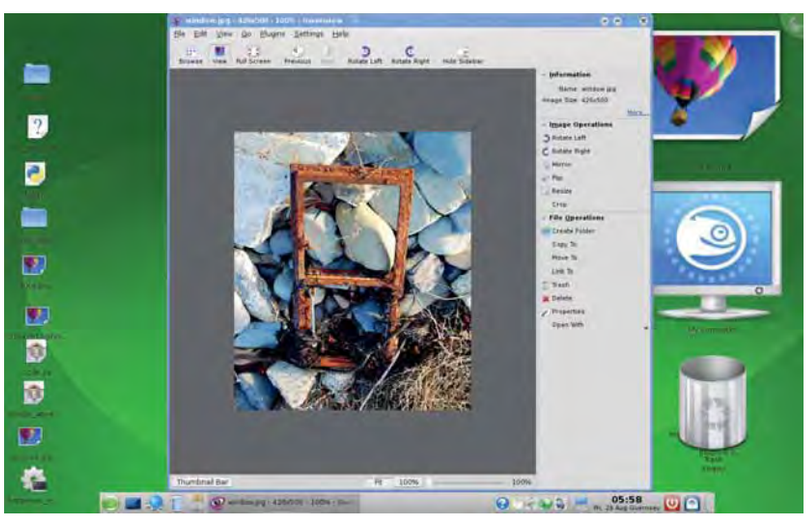
Главное и с ходу заметное преимущество KDE 4.1 – ошеломительная внешность. Но ведь рабочий стол ценят не только и не столько за графику. Да, все верно, но... как тут удержаться, если вас соблазняют эффектами полупрозрачности, сверхплавным рендерингом, современными темами! Заметно, что над ними здорово потрудились, даже после 4.0.

Добавлено много эффектов и изменений для *KWin*, который получил желейные окна и стильный красивый переключатель. Пересмотр *Qt*-библиотек сделал графические эффекты доступнее, а системные требования скромнее. Конечно, весь этот блеск ничего не стоит без прикладных программ, но сам рабочий стол, похоже, к приему приложений готов.

Правда, по части приложений вопросы к новому рабочему столу тоже остались: например, управление файлами, с новым *Dolphin* и старым *Konqueror*. Очень много дублирования. Хотя предполагается, что *Dolphin* будет основным файловым менеджером, а *Konqueror* останется обозревателем Интернета, последний был и остается полноценным файловым менеджером. Вероятно, многие предпочтут *Konqueror*, он привычнее.

Обе программы претерпели значительные обновления: *Dolphin* получил древовидное отображение и вкладки, а *Konqueror* вслед за *Firefox* научился открывать прежние вкладки при запуске и несколько подправил рендеринг.

KDE 4.1 значительно прибавил по сравнению с во многом экспериментальным вариантом. 4.0. Результаты налицо, но, согласны вы с коренным пересмотром интерфейса или нет, несомненно одно: работы предстоит немало. Остается ощущение незавершенности, хотя мы и знаем, сколько труда уже затрачено. В сто раз лучше KDE 4.0, но полную победу праздновать рано. LXF



› ***Gwenview*, смотрелка картинок KDE с бретонским именем, получила упрощенный интерфейс для версии 4.1.**

Грэм считает...

«Не разделяю пессимизма Ника. Все это можно отнести к 4.0, а я уже смог перейти с 3 на 4 – и ничего, работаю.»

LINUX FORMAT Вердикт

KDE 4.1

Разработчик: KDE
Сайт: www.kde.org
Цена: бесплатно под GPL

Функциональность	6/10
Производительность	9/10
Простота использования	9/10
Оправданность цены	8/10

» *Работы осталось полно, но KDE 4.1, по крайней мере, можно пользоваться.*

Рейтинг 7/10

Ankh: Heart of Osiris

Саймон Пикстоск нашел приключения Ассиля в древнем Египте головоломными, и не в хорошем смысле. Обескураживает и сбивает с толку.

Вкратце...

» 3D-приключения с элементами головоломки.

Хотя игра вышла около двух лет назад в Германии, до Британии и нашего журнала она добралась лишь сейчас. *Heart of Osiris* продолжает линию прежних игр *Ankh*, но если вы никогда не встречались с ними раньше, не расстраивайтесь: непонятно только поначалу, предыстория большого значения не имеет. Надо сказать, Linux не избалован играми, полноценно использующими OpenGL – тем более приятно видеть сочную картинку. Однако вы вскоре заметите, что хорошая графика еще не означает хорошую игру.

Установка прошла сравнительно гладко, с примечательным советом отключить *Compiz* – много ли игр, которые о нем вообще знают? Ваш персонаж – Ассиль [Assil], хранитель Анха, хотя этот самый Анх [древнеегипетский символ жизни, – прим. ред.] он ухитрился потерять. Как в любой приключенческой игре, нужно наводить указатель мыши на различные предметы, чтобы выяснить, нельзя ли их употребить. Неблагодарное занятие. Ткнитесь в деревянный ящик, и получите подсказку: «дерево, один из основных предметов экспорта». Может, и смешно, но проку никакого.

Чтобы найти пропавший Анх, придется пообщаться со множеством персонажей, выбирая вопросы из предложенного списка. Разговоры по большей части утомительно однообразны и бестолковы. Стоит задать «не тот» вопрос, как вас бесцеремонно выкинут из дискуссии, и придется начинать все сначала. Но суть игры – в разгадке головоломок, с тем, чтобы получить доступ в новое место или к новому персонажу. Увы, логическая мысль

› **Все головоломки в баре к вашим услугам. Или на вашу голову.**

для головоломок из *Ankh* поможет мало. Например, Ассилю нужно войти в бар, а швейцар не пускает – у него, видите ли, борода.

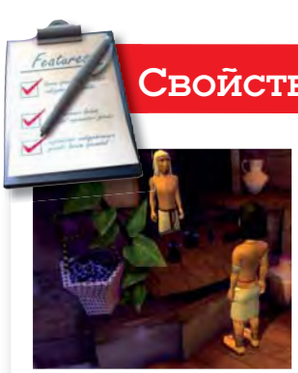
Переголоволомили

Ладно, думаете вы; побреемся. Оглядевшись, видим цирюльника. Но вместо нормального бритья он предлагает удалить бороду с помощью горячего воска. Теперь нужно искать воск; для этого сначала прикупите попугая в клетке и повесьте клетку на крюк перед дверью заброшенной лавки. Голодный кот прыгнет на клетку, сдвинет крюк, на котором она висит, и вы увидите сундучок с воском и печатью. Теперь запихните воск в матрешку – сувенир из детства Ассиля (она припрятана в корзине), попросите огнеглотателя расплавить воск и принесите его парикмахеру.

Кроме обычных головоломок, есть и особые, с использованием кодового диска. Заговорите с продавцом вина, и он предложит приготовить коктейль. Вы соглашаетесь, но тут сценарий дает сбой и вам велят проваливать. А потом заявляют, что забыли рецепт коктейля – мол, не поможете ли? По идее, коктейль нужно приготовить, отмерив нужное количество ингредиентов кодовым диском. Но сколько мы ни пытались это сделать, вместо названий выходили маловразумительные ingredient 1 и ingredient 2. (Странно, но позже в оконном режиме все работало как надо).

Если у вас стальные нервы и нетривиальное мышление, не исключено, что вы и получите некоторое удовольствие от *Ankh*. Однако большинство людей найдет, что игровые проблемы, извращенная логика и нехватка подсказок выводят весь процесс за рамки развлечений. LXF

Свойства навскидку



Головоломки
Помогите продавцу приготовить коктейль, если справитесь с кодовым диском (мы не смогли).

Графика
OpenGL-графика действительно радует глаз, но на акцент «голоса за кадром» лучше не обращать внимания.

LINUX FORMAT Вердикт

Ankh: Heart Of Osiris

Разработчик: Runesof
Сайт: www.ankh-game.com
Цена: ок. $35

Сюжет	4/10
Графика	9/10
Увлекательность	3/10
Оправданность цены	3/10

» *Слишком много раздражающих проблем – тут не до удовольствия!*

Рейтинг 3/10

Opera 9.5

Перемена труда есть отдых – вроде так говорят? **Саймон Пикстон** обнаружил, что смена *Firefox* на *Opera* и впрямь весьма освежает.

Вкратце...

» Браузер, сочетающий все необходимое для работы в Интернете с высокой скоростью загрузки страниц и гибкостью дополнительных модулей. См. также: *Firefox*, *Konqueror*, *Epiphany*.

Среди браузеров для Linux *Opera* всегда была в тени: в KDE встроен *Konqueror*, а Gnome часто предустанавливает *Firefox* – стоит ли возиться еще с одним обозревателем? Но у *Opera* в рукаве есть один козырь: скорость. *Firefox* с каждой версией приобретает все больше функций, но становится неповоротливым. *Opera*, наоборот, нередко многим жертвует во имя главного – производительности. Любой, кто хоть раз пробовал мини-браузер *Opera* на мобильном телефоне, мог это оценить.

Правда, новый установщик ощутимо раздобрел, но скорость по-прежнему во главе угла, и *Opera 9.5* безусловно превосходит старые версии. В основном за это следует благодарить новый движок рендеринга, хотя *Opera* заявляет, что сторонние расширения теперь тоже прибавили в скорости.

Если вам когда-нибудь приходилось искать важный кусок информации, на который вы забыли поставить закладку, вы помните, как нудно рыться в журнале в попытках набрести на след. Quick Find (быстрый поиск) – очень полезная вещь: запоминается не только заголовок и адрес страницы, но и ее содержание. Функция работает в связке с адресной строкой: достаточно ввести ключевое слово – и вы получите не только список адресов, где побывали, но и страницы с этим словом.

› Функция Speed Dial сопровождает каждую новую вкладку, предоставляя наглядную ссылку на девять ваших любимых web-сайтов.

> «У Opera всегда имелся хотя бы один козырь в рукаве: это скорость».

Opera, вперед!

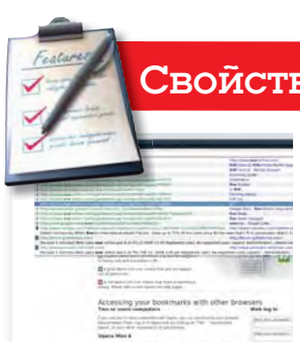
Если *Opera* установлен у вас на мобильнике, или просто на нескольких ПК, вам понравится функция Opera Link – интеллектуальное средство синхронизации закладок между всеми имеющимися у вас экземплярами браузера. Весь комплект закладок можно без хлопот передать не только с ПК на мобильный, но и наоборот (удобно). Еще одна новинка – Speed Dial (быстрый доступ). Это представление девяти любимых сайтов в виде таблички миниатюр «три-на-три» – удобный способ мгновенно увидеть реально востребованное. Такая табличка появляется при открытии каждой новой вкладки. Если синхронизировать с мобильным устройством, то вместо миниатюр вы увидите девятку ссылок и мгновенно перейдете на нужный сайт без необходимости вводить адрес.

Стоит напомнить, что *Opera* – также и полноценный почтовый клиент, а встроенный клиент BitTorrent пулей загрузит для вас любой Linux-дистрибутив.

Как и в *Firefox*, можно загружать и устанавливать дополнительные модули в форме виджетов. Жесты мышью доступны из коробки, так что можно вовсе не выпускать вашего пластикового грызуна из рук. Главная забота многих пользователей – безопасность, и *Opera* предлагает надежную защиту от фишинговых сайтов (включено по умолчанию).

Opera 9.5 – не революция, но браузер имеет солидный фундамент, а скорость по-прежнему непревзойденная (исключая, может быть, *Lynx* с аналогами). Хотя большинство новых функций вынесено в добавочные модули, причин для смены версии предостаточно. LXF

Свойства навскидку



Найди ЭТО
Quick Find – контекстно-зависимый поиск с учетом не только названий сайтов, но и их содержимого.

Выбирайте виджеты
Как и *Firefox*, *Opera 9.5* предлагает виджеты для добавочной функциональности.

LINUX FORMAT Вердикт

Opera 9.5

Разработчик: Opera
Сайт: www.opera.com
Цена: бесплатно для некоммерческих целей

Функциональность	10/10
Производительность	10/10
Простота использования	9/10
Оправданность цены	9/10

» *Ряд инноваций и впечатляющая скорость приглашают внимательно присмотреться к новой версии.*

Рейтинг 9/10

Synology DS207+

Грэм Моррисон любуется новеньким сетевым накопителем под управлением Linux: все в лампочках, черная панель управления и белый пластик корпуса. Кто сказал «восьмидесятые»?..

Вкратце...

» До 2 ТБ RAID-хранилища под управлением встроенного Linux. Среди недорогих альтернатив – Excito Bubba, QNAP 109 Pro, QNAP 409 и, конечно, гость прошлого номера Linutop.

СПЕЦИФИКАЦИЯ

CPU: 500 МГц ARM926EJ

ЯДРО: 2.6.15

ОЗУ: 128 МБ

НАКОПИТЕЛИ: два 3,5-дюймовых диска SATA + разъем для внешнего SATA

РАЗЪЕМЫ: 2 x USB, Gigabit LAN, печать USB CUPS

› Способность работать с двумя накопителями позволяет DS207+ войти на рынок малых предприятий. Для них это, возможно, идеальный выбор.

Недавний онлайн-опрос показал, что немалая часть наших читателей планирует покупку сетевого хранилища (NAS). Нетрудно понять, зачем. Эти удобные устройства постепенно становятся центрами хранения информации в домашних сетях и в небольших офисах. Разумеется, мы не могли пройти мимо – за последние 12 месяцев охватили не одно такое. Большинство из них работает на Linux, отчего получается весьма привлекательная энергосберегающая замена настольным ПК. Только вот глубина, на которую устройства прячут Linux, у всех разная.

Без жестких дисков, корпус DS207+ на удивление легкий. Два маленьких винта скрепляют половинки: стоит их отвернуть, как внешняя оболочка разъезжается на две части. Внутри стандартная металлическая этажерка под два диска, два SATA-разъема и два разъема питания. В комплект входят короткие кабели питания и данных для подключения накопителей. Рекомендуем сначала соединить разъемы, и только потом крепить диски к кронштейнам. Подключаем блок питания и сетевой кабель – и готово? Не совсем.

А где Linux?

По умолчанию, операционной системы на устройстве нет, а инсталлятор работает только из-под Windows и Mac OS X. Это непростительно, ведь устанавливать-то нужно Linux! Мы решили проблему так: оказалось, инсталлятор нормально работает под *Wine*, если скопировать каталог **DSAssistant/Application** с CD-ROM в *Wine*-папку **drive_c/Program Files**. Приложение отыщет NAS-устройство и установит туда прошивку (она тоже на CD-ROM). Весь процесс занял минут пять, и вскоре мы распростились с *Wine* и вошли в NAS через web-интерфейс. Возня с совместимостью платформ осталась позади.

› DS207+ ведет себя тихо: все внутренности охлаждаются единственным кулером. Можно поручить ему функцию наблюдателя, подключив сетевые камеры.

Web-интерфейс – самый удобный из всех NAS, которые мы когда-либо видели, а встроенная Linux-система снабжена серьезными функциями. Первым делом нужно настроить тома хранилища. Имея два диска, можно выбирать между RAID 0, 1 и 5, настройка несложная. Идем дальше. Для игровых консолей имеем потоковую передачу DNLA, хостинг для музыки – *iTunes*. AMP, FTP и файловые серверы – все это вызывается буквально парой щелчков мыши. Серверы – смесь открытых решений, собственных Synology-разработок для DNLA-стриминга и работы с фотоальбомами, и аутентификации в Active Directory для Windows-пользователей.

Мы нашли производительность DS207+ отменной, а подключаемость к другим устройствам – безупречной. Более того, настроив SSH, мы получили полный доступ к ОС, смогли установить инструментарий программиста и скомпилировать собственные пакеты. DS207+ – отличный вариант, и работает гладко. Хотелось бы только, чтобы разработчики Synology уделили больше внимания Linux-клиенту, раз уж они освоили «прошивку» Linux. LXF

LINUX FORMAT Вердикт

Synology DS207+

Разработчик: Synology
Сайт: www.synology.com
Цена: ок. $450

Функциональность	9/10
Производительность	8/10
Простота использования	5/10
Оправданность цены	7/10

» *NAS, не зависящий от ОС, решение хорошее. Но никаких поблажек Linux-пользователю; а жаль.*

Рейтинг 7/10

Runtu 3 Desktop

Пусть кому-то это и не нравится, но Ubuntu – нынешний фаворит мира дистрибутивов Linux. Можно ли сделать его еще лучше? **Василий Шингур** пробует разобраться.

Вкратце...

» Настольный дистрибутив для русскоязычных пользователей. См. также: ALT Linux, Mandriva или оригинальный Ubuntu.

Ключевое ПО

» Ядро 2.6.24-21-generic
» X.Org 7.3
» Gnome 2.22
» OpenOffice.Org 2.4.1 Pro
» Firefox 3.0.3
» Thunderbird 2.0.17
» Pidgin 2.5.0
» Wine 1.1.5
» GIMP 2.4.6
» Мультимедиа-кодеки

1 октября 2008 года на российский рынок вышел дистрибутив Runtu 3 Desktop, базирующийся на Ubuntu 8.04.1 и созданный силами команды разработчиков Runtu.org. Выбор фундамента для Runtu 3 легко объясним: это стабильный LTS-релиз Ubuntu, который будет поддерживаться вплоть до мая 2011 г. Ожидание пользователей длилось долго: разработчики намеренно отказалась от поспешности в пользу надежности. Выход Runtu 3 оттягивало и стремление вывести систему на один уровень с ведущими мировыми дистрибутивами. В итоге команде пришлось решать сразу три задачи: создавать продукт, получать лицензию и организовывать техническую поддержку.

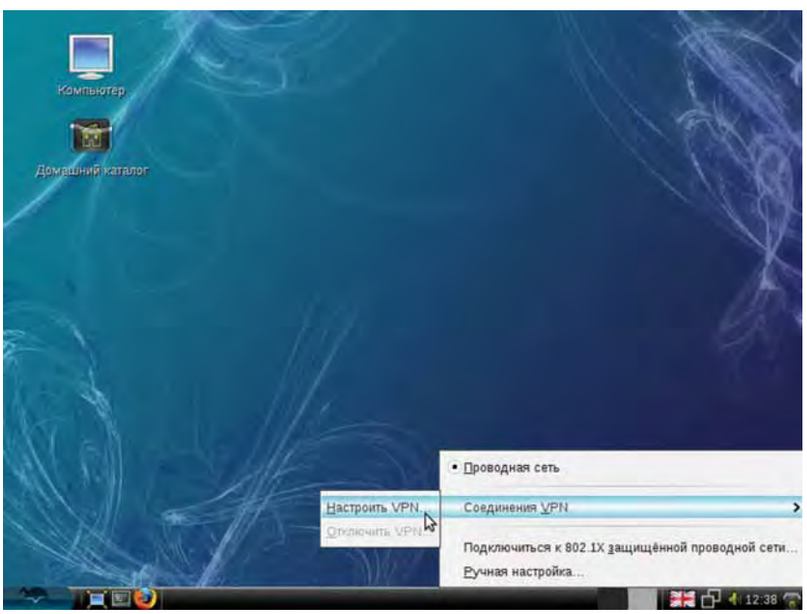
Runtu 3 доступен в следующей комплектации: DVD-диск с ОС для архитектуры x86, буклет с краткой инструкцией по установке, лицензия и купон технической поддержки сроком на один месяц с момента первого обращения. Лицензия позволяет устанавливать Runtu 3 Desktop на неограниченное число компьютеров, однако поддержка по одному купону осуществляется только для одного ПК. Разумеется, дистрибутив можно также свободно загрузить с официального сайта проекта – он распространяется на условиях GNU GPL.

Под капотом

В состав этой версии Runtu вошло множество обновлений и дополнительных программ, которые расширили функциональность базовой системы с учетом отечественной специфики.

Так, для работы с документами в Runtu 3 Desktop предустановлен пакет *OpenOffice.org Pro 2.4.1* от компании «Инфра-Ресурс», ориентированный на русскоговорящих пользователей. Не забыть о важных событиях позволяет календарь-планировщик *Sunbird*.

Целевая аудитория Runtu – люди, не имеющие дешевого широкополосного доступа к Сети, но это не значит, что его «среднестатистический пользователь» будет существовать в отрыве от остального мира. В состав системы включены не только привычный web-браузер *Mozilla Firefox 3* и почтовый клиент *Mozilla Thunderbird 2*, но и утилиты для настройки сетевого соединения, будь то популярный у российских интернет-провайдеров VPN (PPTP/L2TP) или же мобильный телефон с поддержкой GPRS. Благодаря инструментам *Gnome PPP* и *NetworkManager* подключение к Интернету займет не более двух минут. Для онлайн-общения есть мессенджер *Pidgin*, а для обмена файлами – BitTorrent-клиент *Deluge*, *DC++* и удобный менеджер закачек *D4X*.



› **Runtu поможет настроить выход в Интернет, не осаждая телефон службы поддержки провайдера.**

Система подготовлена и в плане мультимедиа: здесь есть универсальный видеопроигрыватель *Totem*, аудиоплейер *Audacious* с набором аудио- и видеокодеков; графический редактор *GIMP*, фотоальбом *F-Spot*. Уделено внимание и интеграции с Windows и Win32-приложениями: в стандартный комплект поставки включены *WINE* и *Samba* (инструмент обмена файлами в сетях Microsoft).

К сожалению, из финальной версии Runtu 3 были исключены проприетарные драйвера для видеокарт NVIDIA и ATI, а также пакет *Unrar*, лицензия которого является несвободной. Это несколько снижает ценность дистрибутива для оффлайновых пользователей, но разработчики обещают как-то исправить ситуацию. По тем же лицензионным соображениям были удалены и подключаемый модуль Adobe Flash наряду со средой Sun Java: им на смену пришли свободные Gnash и OpenJDK.

С заделом на будущее

Конечно, с выходом Runtu 3 развитие проекта не остановилось. Команда готовит специализированные дистрибутивы: Runtu 3 Office для слабых офисных и домашних ПК, Runtu EEE для пользователей субноутбуков Asus EeePC и Runtu Flash – «живую» USB-редакцию. В разработке находится Runtu Server, ориентированный на серверные функции. Его изюминкой станет визуализация основных серверных функций.

Из других нововведений следует отметить серию Runtu Services; ее выход запланирован на 1-й квартал 2009 г. По заявлениям разработчиков, Runtu Services – это наборы готовых стеков ПО, предназначенные для тех, кто не может позволить себе скачивать крупные наборы данных. Ожидается, что в это семейство войдут 4 продукта: Server Service, Office Service, Student Service и Multimedia Service. Как говорится, следите за новостями. LXF

LINUX FORMAT Вердикт

Runtu 3 Desktop

Разработчик: Команда Runtu
Сайт: www.runtu.org
Цена: 200 руб. или бесплатная загрузка

Функциональность	7/10
Производительность	8/10
Простота использования	8/10
Оправданность цены	9/10

» *Если коричневый – не ваш цвет, а необходимость скачивать пакеты из Интернета раздражает, попробуйте это.*

Рейтинг 8/10

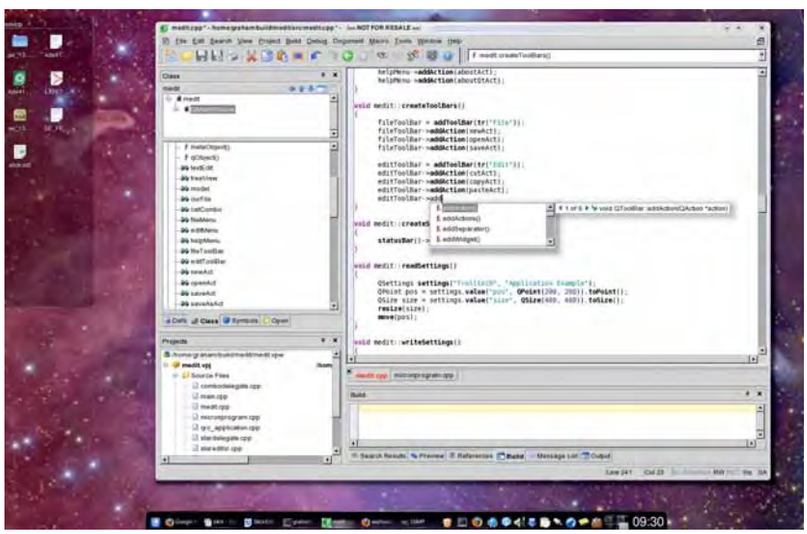
SlickEdit 2008

Выложив почти 300 баксов, **Грэм Моррисон** осознал, что «гладкое редактирование» – это не только *Vim* и шелковые кашне...

Вкратце...

» Текстовый редактор, переросший в интегрированную среду разработки, с автозавершением, подсветкой синтаксиса, встроенным компилятором и отладчиком. Среди бесплатных альтернатив – *XEmacs*, *KDevelop* и *Eclipse*.

Для программиста текстовый редактор – это как пара хорошо растоптанных походных ботинок. Где узко – разношено, лишнее урезано «под себя»: процесс подгонки длится годами. Редактор переживает с программистом и взлеты, и падения – бессонные ночи и ранние подъемы. Последнее дело, если вдруг среди ночи редактор «заест». Именно поэтому программисты столь разборчивы, и именно поэтому недешевый редактор *SlickEdit* существует столько лет, поднявшись от командной строки к нынешнему графическому варианту. Минуло два года с тех пор как мы обращались к *SlickEdit* последний раз, и за это время пропустили вариант 2007. Но потеряли немного. *SlickEdit 2007*, по общему признанию, страдал множеством ошибок (по крайней мере, первые несколько версий). Кроме того, лишь недавно была пересмотрена запутанная процедура онлайн-регистрации.



› ***SlickEdit*** **все больше напоминает IDE, только намного проще и быстрее аналогов типа Eclipse.**

Свежесть перемен

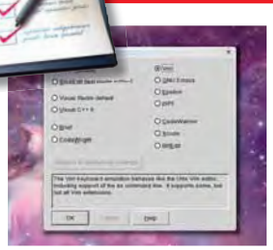
SlickEdit 2008 – это скорее интегрированная среда разработки (IDE), чем просто текстовый редактор. В нее включены отладчик, браузер классов, интеллектуальный ввод текста с автозавершением – все это автоматически настраивается при установке приложения. Используя новую систему импорта Make-файлов, мы смогли мгновенно загрузить несколько текущих проектов, без промедления продолжив работу и отладку – точно так же, как в *KDevelop* или *Eclipse*. Единственное, чего не хватает – микроменеджмента зависимостей в Make-файлов. Возможен импорт проектов из других IDE (например, борландовского JBuilder или Xcode от Apple), что весьма полезно при коллективной разработке. Правда, с загрузкой Xcode-проекта у нас были проблемы.

Однопользовательская лицензия, выдаваемая при покупке *SlickEdit*, дает право работать на семи разных платформах (включая Windows, OS X, Solaris и, естественно, Linux), а с новыми функциями, такими как адаптивное форматирование, программа будет приветствоваться на любой ОС: эта функция, анализируя текстовый буфер, автоматически следит за интервалами табуляции и прочими элементами стиля форматирования, что очень полезно при мелкой подчистке чужих проектов. Ни в одном из свободных текстовых редакторов, к которым мы привыкли, ничего подобного нет. Есть также очень полезный инструментарий с блестящей функцией «украшения», средствами отслеживания изменений и оптимизации; все это значительно упрощает работу над исходным кодом. Сделанные изменения можно зафиксировать в различных системах контроля версий, не покидая *SlickEdit*.

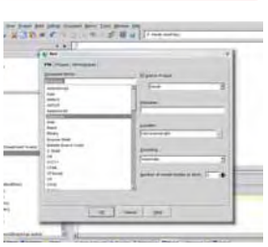
Ошибки в окне

Еще одно ценное новшество – панель Message List [Список сообщений], которую в *KDevelop* мы давно принимаем как данность. Она собирает все обнаруженные ошибки в единственном окне, и двойной щелчок на предупреждении тут же переносит нас к проблемной строке. Есть и дополнения помельче, например, генератор скобок, транслятор символов и новая панель опций. Хотелось бы видеть встроенную API-документацию и средство работы со словарем, а также более плотную интеграцию с двумя ведущими GUI-библиотеками Linux и визуальное свертывание кода. Но сила *SlickEdit* (и причина, по которой программисты охотно за него платят) – это скорость и гибкость. Программа работает с ошеломляющим количеством языков программирования, и работает стремительно. Для программиста-полиглота, ценящего удобство и комфорт рабочей среды, *SlickEdit* 2008, вероятно, единственный вариант. LXF

Свойства навскидку



Эмуляция
Можно настроить *SlickEdit* на эмуляцию клавиатурных команд различных текстовых редакторов, включая последнюю версию *Vim*.



Полиглот
В *SlickEdit* возможен анализ, загрузка и подсветка более чем 80 различных текстовых форматов.

LINUX FORMAT Вердикт

SlickEdit 2008

Разработчик: SlickEdit, Inc
Сайт: www.slickedit.com
Цена: $299

Функциональность	9/10
Производительность	9/10
Простота использования	8/10
Оправданность цены	6/10

» *Слишком дорого для единственной платформы и одного языка программирования. Но у программиста-полиглота мигом окупится.*

Рейтинг 8/10

В «Системном администраторе» вы не прочтете о:

– котировках валют
– сплетнях
– погоде
– политике
– развлечениях

В вашем распоряжении:

– опыт лучших IT-специалистов
– новые идеи и полезные советы
– самые эффективные решения в области системного и сетевого администрирования

Подпишитесь сейчас!

Роспечать – 20780, 81655
Пресса России – 87836
Online-подписка – www.linuxcenter.ru

Время подписки ограничено!

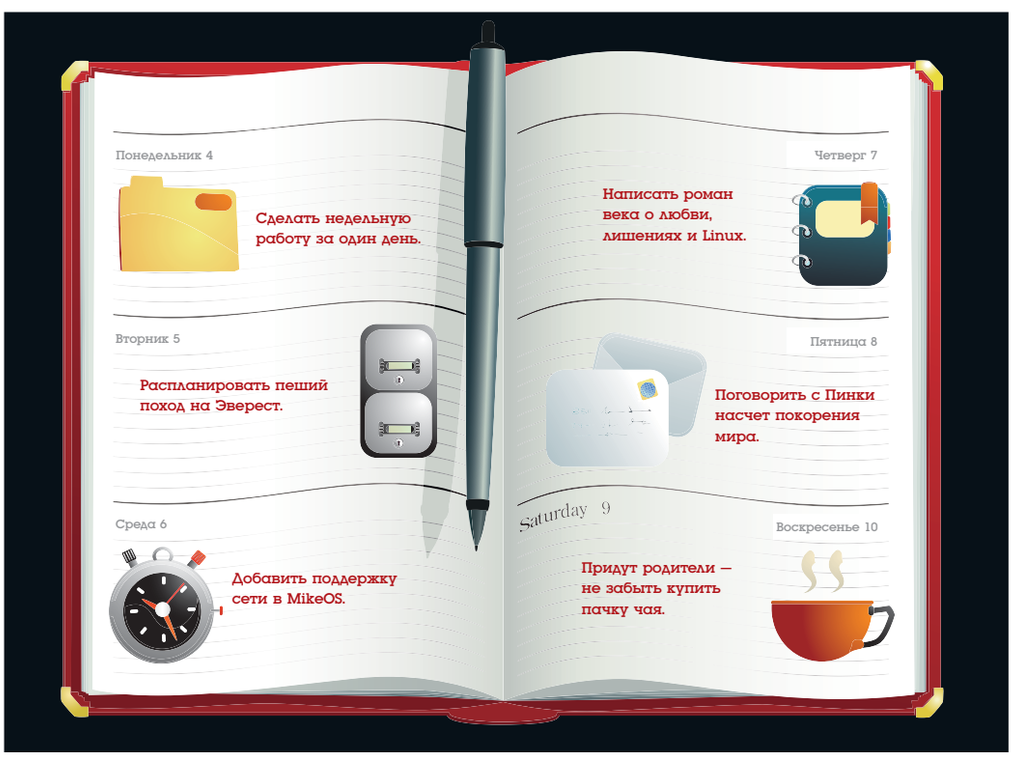
Сравнение

» Каждый месяц мы сравниваем тысячи программ — а вы можете отдыхать!

LINUX FORMAT На DVD

Как привести дела в порядок

Завалены заданиями, to-do-списками и встречами, которые прут на вас изо всех углов? **Марко Фиоретти** изучает, как раскидать их с помощью GTD-инструментов Linux.



Про наш тест

Все приложения, рассмотренные в этом Сравнении, проверялись на системе AMD64 3800+ с двухъядерным процессором и 4 ГБ ОЗУ, под управлением Fedora Core 8.

Сначала мы смотрели, насколько быстро и легко установить каждую программу в систему; затем проверяли соответствие официальному методу ДВП и дружелюбие интерфейса.

Скорость и простота интерфейса отражаются в легкости ввода задач в систему, так что они весьма важны. Гуру ДВП постоянно повторяют, что для успешного использования методологии вам следует использовать... да все, что угодно – лишь бы оно было быстрым, не путалось под ногами и не требовало больших раздумий!

Ну и, наконец, мы оценивали, насколько стабильно каждое приложение и как оно интегрируется в рабочий стол, начиная с взаимодействия с различными почтовыми и календарными клиентами.

Наш выбор

Basket с. 24
Chandler с. 23
Emacs Org-mode с. 21
GTD-Free с. 21
Kontact с. 24
MonkeyGTD с. 22
ThinkingRock с. 23
Tracks с. 22

Привлечение приложений по управлению временем для организации вашей деятельности – хорошая мысль, если только это не станет еще одним препятствием на пути к цели Getting Things Done (GTD – ДВП, Дела в порядке). Сокращение, данное Дэвидом Алленом для своего метода управлением временем, стало хорошим знаком. Кроме того, приверженцев у GTD куда больше, чем у *GNU Emacs*. Общая мысль такова: выгрузите из головы все, что вам нужно сделать, перенеся это в надежную систему, основанную на очередности действий, регулярных анализов и «памяток», которые ничего не забывают и волшебным образом показывают, что вам надо делать сейчас. Так вы сильно увеличите свою производительность.

Памятка – это любой физический инструмент или приложение, который хранит напоминания и сидит в тени до тех пор, пока не настанет пора озаботиться о нем; тут он и появится. Узнайте больше о GTD на www.43folders.com/2004/09/08/getting-started-with-getting-things-done.

Луддитский метод ДВП не использует ничего, кроме ручки и бумаги – и миллионам людей этого хватает. Но истинный линуксоид просто обязан хоть раз испробовать цифровую альтернативу к любой работе; и вот вам восемь программ, предназначенных как чисто для ДВП, так и для близких задач.

> «Выгрузите из головы все, что нужно сделать, в надежную систему.»

Мы подбирали однопользовательские приложения, способные работать оффлайн, чтобы отслеживать действия и группировать их по проектам, категориям или контексту (а именно, местоположение или инструмент для выполнения чего-нибудь, например, «Офисный компьютер» или «Телефонные звонки»). Учитывались также интеграция с почтовым клиентом и календарем, поддержка распечатки и переносимость данных.

Emacs Org-mode

И Emacs не чуждо желание ДВП!

Мы отметили, что для ДВП требуется наивозможно быстрый и простой интерфейс ввода. Но что может быть проще и быстрее ввода простого текста?

Ага, ощутили охотничий азарт? Оно и понятно: шаг от этой мысли до добавления в ДВП-режима Org (от organisation) для символьной операционной системы, известной как *Emacs*, настолько очевиден, что его давным-давно уже сделали.

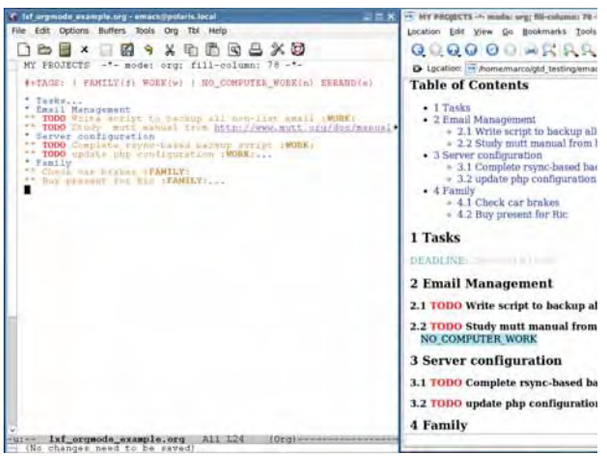
Пользовательский интерфейс – стандартный *Emacs*'овский: одно выпадающее меню и масса новых пальцеломных сочетаний клавиш. Если вас это не страшит, то в *Org-mode* (включенном в *Emacs 22*) можно сохранять заметки, управлять проектами, составлять расписание встреч и многое другое. Специализированные функции быстро создают, помечают тегами, фильтруют и отображают иерархические to-do-списки с галочками. Другие макросы временно прячут от ваших глаз менее срочные задачи.

Если вы помещаете куски текста в одну строку и разделяете их символом канала (|), Org-mode автоматически переформатирует их в выровненную таблицу. Можно также создавать расписание в HTML и *LaTeX*.

Разметка очень проста: звездочки создают маркеры и субмаркеры; гиперссылки на сайты и локальные файлы определяются автоматически; «навешивание» тега осуществляется вводом его имени, обрамленного двоеточиями. Имеются специальные теги: например, CLOCK отслеживает, сколько времени вы потратили на каждую задачу.

Явные преимущества простого текста для ДВП отрицать сложно: вы можете работать в любом текстовом редакторе на любой операционной системе безо всяких отвлекающих вас прибамбасов. Кроме того, есть отличная документация, а статьи вроде **http://members.optusnet.com.au/~charles57/GTD/orgmode.html** детально освещают, как использовать ДВП в Org-mode.

Единственной проблемой *Org-mode* является, конечно, сам *Emacs*! Если вы не умеете или не хотите им пользоваться, испытайте ДВП в чем-нибудь другом.



› ***Emacs*, конечно, устрашает, но правила *Org-mode* обманчиво просты, и вы можете создавать HTML- или PDF-отчеты.**

LINUX FORMAT Вердикт

Emacs Org-mode

Версия: 6.06b
Сайт: http://orgmode.org
Цена: Бесплатно

» *Если вы уже используете Emacs или просто любите текстовые интерфейсы, есть смысл остановиться на Org-mode.*

Рейтинг 7/10

GTD-Free

Быстр и прост в использовании, но не особо стабилен.

GTD-Free – легковесная Java-программа: чтобы запустить ее, скачайте с сайта последнюю версию в формате JAR, скопируйте, куда вам удобнее, и введите в терминале следующее:

```
java -jar gtd-free-0.3.1.jar
```

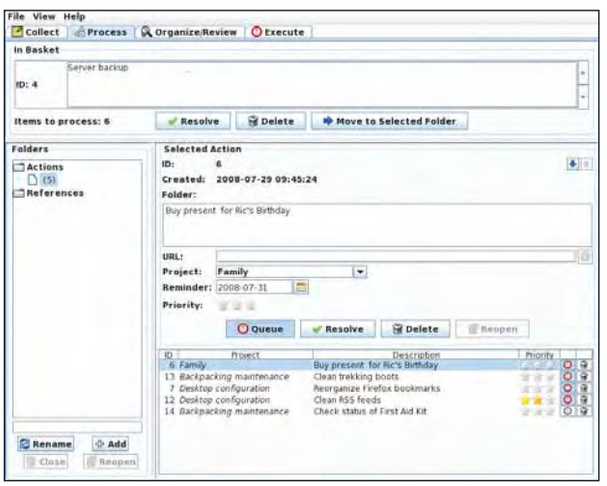
В верхней части наглядного интерфейса, сразу под главным меню, находятся четыре вкладки, соответствующие главным фазам управления задачами в ДВП: Сбор [Collect], Обработка [Process], Организация/Обзор [Organise/Review] и Выполнение [Execute].

Интерфейс ввода задачи во вкладке Collect, пожалуй, является самым быстрым среди всех приложений этого Сравнения: просто вводите все, что ни попадя, жмите Enter, вводите следующее задание, затем снова Enter; и так вплоть до полной разгрузки вашего мозга в *GTD-Free*. Все набранное попадает в корзину «In» в нижней части окна. Для вашего спокойствия, данные сохраняются в одном XML-файле каждые несколько минут.

Вкладка Обработка [Process] – это то место, где вы изначально создаете папки для действий и справочных материалов, согласно методу ДВП. Однако большую часть проводимого здесь времени вы потратите на обработку ранее собранных действий. Во-первых, переместите их в правую (под)папку Действия [Actions] или Ссылки [References]. После этого нажатием нескольких клавиш назначьте проекту действия и напоминания. Если вы хотите, чтобы текущее действие отобразилось в очереди Следующего действия [Next Action], нажмите соответствующую кнопку.

Периодическое рассмотрение текущей ситуации происходит в панели Организация/Обзор [Organise/Review], где предлагается несколько представлений для незавершенных заданий. Панель разделена на три части: одна для Папок [Folders], другая для отображения или фильтрации очереди Следующего действия, а третья – для проверки или редактирования деталей выбранного действия. В последней панели настает время браться за дело. Сядьте, выполните то, что вам велит *GTD-Free*, а затем с чувством выполненного долга удалите это из системы!

GTD-Free имеет меньше возможностей, чем конкуренты (а именно, нет интеграции с электронной почтой), но он быстр, прост, легок в освоении и переносим. Единственная проблема, которую мы обнаружили – частые зависания, в особенности при переходе между панелями Обработкой [Process] и Организацией [Organise]. Именно по этой причине мы пока не рекомендовали бы эту программу.



› ***GTD-Free* предлагает ДВП в дружественном интерфейсе.**

LINUX FORMAT Вердикт

GTD-Free

Версия: 0.3.1
Сайт: http://gtd-free.sourceforge.net
Цена: Бесплатно

» *Возможно, это самый простой и быстрый графический ДВП для Linux, но данная версия не рекомендуется из-за нестабильности.*

Рейтинг 6/10

MonkeyGTD

ДВП, внедренный в однопользовательскую wiki.

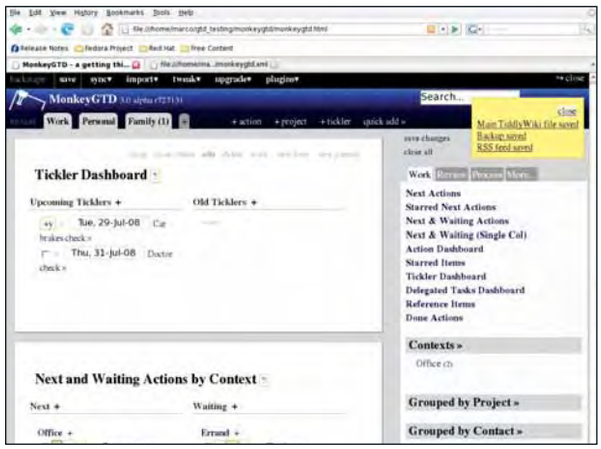
Существует интересная категория утилит ДВП: wiki, состоящие из единственной HTML-страницы. Обычно они обвешаны шаманством типа JavaScript и CSS, для сокрытия или отображения, по вашей команде, любого сочетания введенных вами данных. Такие приложения прекрасно работают без соединения с Интернетом.

Существует несколько подобных wiki, но у нас нашлось место только для одного варианта: следуя заголовку этого Сравнения, мы выбрали тот, чья раскладка по умолчанию всего привычнее адептам чистого ДВП. Хотя во многом это справедливо и для *GTDTiddlyWikiPlus* (**www.checkettsweb.com/tw/gtd_tiddlywiki.htm**).

Область Backstage Area вверху предназначена для настройки, системного администрирования и синхронизации с удаленной установкой *MonkeyGTD*. Для повышения безопасности можно также блокировать редактирование, когда к *MonkeyGTD* обращаются из Интернета.

Если вы раньше не работали с wiki, изучение простейшей разметки потребует времени, но после этого Привести дела в порядок по книге с помощью *MonkeyGTD* очень просто. Одним щелчком отображаются все активные проекты без Очередного действия [Next Action], а также Помеченные [Starred], Ждущие [Waiting] и Делегированные [Delegated] действия, пункты Ссылки [Reference items] и Доска памяток [Tickler Dashboard]. Поддерживаются метки, подпроекты и периодические события. Четкие «Сферы» [Realms] (Work, Personal и Family на экранном снимке) отделяют друг от друга разные аспекты вашей жизни. При сохранении изменений помните об обновлении RSS-ленты: чтение ее из *Kontact* или *Evolution* может сильно увеличить полезность *MonkeyGTD*.

Благодаря JavaScript, *MonkeyGTD* довольно быстр, даже при использовании окна поиска справа вверху. Некоторое (легкое) раздражение вызвали мерцание при открытии и закрытии панели и тот факт, что *Konqueror* (но, на наше счастье, не *Firefox*) спрашивал подтверждение для каждого cookie, которое он получал из *MonkeyGTD*, хотя и был настроен на «прием всех cookie из этого домена».



› ***MonkeyGTD*** **имеет все необходимые компоненты системы Getting Things Done, в понятном, приятном формате.**

LINUX FORMAT Вердикт

MonkeyGTD

Версия: 3.0 alpha r727131
Сайт: http://monkeygtd.tiddlyspot.com
Цена: Бесплатно

» *Wiki ДВП, которую можно использовать даже на ходу – коль скоро вы озаботились изучить разметку.*

Рейтинг 7/10

Tracks

Многоплатформенный, многопользовательский сервер.

Девизом *Tracks* является «Все Делать Правильно» – он соблюдает принципы ДВП весьма точно, но не настолько истово, чтобы потерять в гибкости. Технически говоря, *Tracks* – приложение Ruby on Rails, работающее в связке с web-сервером *Webrick*. *Tracks* кросс-платформенный и доступен как оффлайн, так и через Интернет.

Кроме Ruby, *Tracks* требует модулей расширения Ruby для выбранной вами базы данных: *SQLite* 3, *MySQL* или *PostgreSQL*; рекомендуем первую, поскольку ее настройка проще и не требует никакого другого сервера. Интерфейс Ajax, стоящий по умолчанию, понятный и быстрый: альтернатива для мобильных телефонов доступна на **http://tracks.default.site/mobile**.

Процедура установки требует некоторого внимания: распакуйте архив, пропишите тип выбранной базы данных в файл **config/database.yml**, убедитесь, что настройки в **config/environment.rb** соответствуют вашей системе, и запустите сервер с помощью:

```
$ABSOLUTE_PATH_TO_TRACKS_ INSTALL_DIR/script/server -e production
```

Наконец, откройте **http://0.0.0.0:3000/signup** в браузере, настройте необходимые учетные записи администратора и пользователя, и – вперед, *Tracks*!

Нам предоставляются Действия [Actions], Контексты [Contexts] и Проекты [Projects]. Каждый Проект имеет свою страницу, где перечислены отложенные и активные Действия. Можете добавлять заметки – и для всего Проекта, и для отдельных Действий. Последние можно помечать тегами или звездочками, когда они действительно важны, а также фильтровать по Контексту, Проекту или Тегам [Tags]. Действия со звездочкой имеют собственную панель. Естественно, в любом представлении можно вернуться от Действия обратно к его Проекту. *Tracks* также интегрируется с любым календарем или RSS-читалкой.

Tracks хорош для отслеживания дел и поощрения пользователя: специальные вкладки покажут, что завершено за последний день, месяц или год, или среднее время выполнения задач. Вдобавок к сохранению данных в форматах YAML, CSV или XML, вы можете разговаривать с *Tracks* на нескольких языках программирования. Таковыми являются, например, скрипты Perl и Ruby, которые вставляют повторяющиеся задачи через *Cron*. В общем, это прекрасная программа!

› ***Tracks*** **передает ваши действия любому клиенту iCal или RSS.**

LINUX FORMAT Вердикт

Tracks

Версия: 1.6
Сайт: www.rousette.org.uk/projects
Цена: Бесплатно

» *Многопользовательский, простой, полноценный и приверженец ДВП, но без фанатизма. Tracks хорош!*

Рейтинг 8/10

Chandler

Органайзер и почтовый клиент.

Chandler – очень интересное приложение, часть более крупного проекта, рекламируемого как «записная книжка, которую можно организовать, резервировать и коллективно использовать». Здесь мы рассмотрим только его Python-клиента для настольных систем.

Хотя визуально ДВП в *Chandler* не бросается в глаза, есть сведения об использовании его в данном качестве (**http://chandlerproject.org/Projects/UserStories**), и более пристальный взгляд убеждает, что это так, если не цепляться за терминологию.

ДВП-проекты? Согласно давним пользователям *Chandler* – а есть ли лучший источник? – наиболее эффективной их реализацией в *Chandler* будет создание по пункту на проект и ввод в него всех «очередных действий». Чтобы облегчить себе жизнь, действиям можно придать подходящие префиксы. Также можно присваивать пунктам теги или сортировать их по Коллекциям [Collections], которые различаются по цветам и которые можно использовать как Контексты ДВП [GTD Contexts] или как хранилище для Справочного материала [Reference material].

Внутри пункта, *Chandler* распознает ссылки как на web-страницы, так и на локальные файлы, каталоги или приложения. Можно использовать эту функцию для планирования действий с записями в календаре, альбомами изображений или документами, которые надо не забыть отредактировать.

В календаре *Chandler* вы можете быстро вставить события в расписание – с повторами или без – и совместно использовать с другими пользователями. Однако центром интерфейса *Chandler* является специальная Коллекция, которая называется Доска [Dashboard] (для отображения ее содержимого просто щелкните по ее названию в левой панели).

Самое ценное свойство *Chandler* – способ его интеграции с электронной почтой, списком задач и встреч. Он просто отображает все в текущей коллекции как единый список пунктов. Чтобы интеграция с почтой заработала, у вас должна иметься рабочая учетная запись IMAP, и *Chandler* нужно иметь возможность создать в ней свои папки. Если для вас это не проблема, *Chandler* будет вам весьма полезен.

› *Chandler* экономит ваше время, отображая заметки, почту и встречи в одном окне.

LINUX FORMAT Вердикт

Chandler

Версия: 1.0-rc1
Сайт: http://chandlerproject.org
Цена: Бесплатно

» *Chandler – не чистый ДВП, но объединяет заметки, календари и почту очень интересным способом.*

Рейтинг 7/10

ThinkingRock

Чистый ДВП в кросс-платформенном пакете Java.

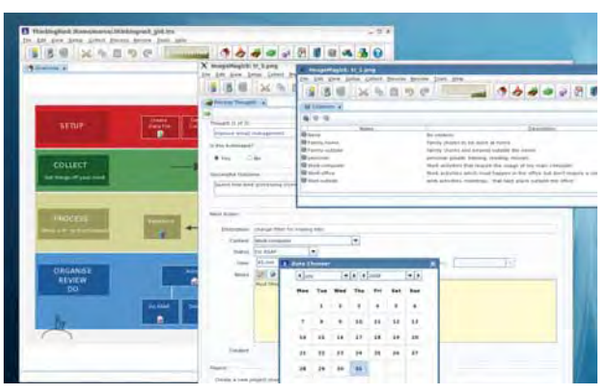
Из всех представленных здесь пакетов, в *ThinkingRock* каждый бит прямо-таки разит ДВП. Начальное окно выглядит как его официальная блок-схема: Сбор [Collect], Обработка [Process] и Организация/Обзор/Выполнение [Organise/Review/Do]. Как и *GTD-Free*, *ThinkingRock* – приложение Java. Для запуска под Linux разархивируйте его и запустите имеющийся скрипт командной строки:

```
./bin/tr.sh
```

Хорошим (или плохим?) качеством *ThinkingRock* является то, что он безотлучно с вами нянчится. Сообщив *ThinkingRock*, где хранить ваши данные, вы должны определить Контексты [Contexts]. Функция Автоматического упорядочения [Automating Sequencing] очень эффективно ранжирует ваши Действия [Actions], но ей нужно задать детальные критерии. Что для вас важнее: Время [Time], Затраты [Energy] (от Никаких [None] до Высоких [High], Умственные [Mental] или Физические [Physical]) или же Приоритет [Priority] (Must, Should, Would, Could)? Затем вы должны назвать интересующие вас темы и назначить цвет каждой из них.

Создание действия состоит не просто в заполнении предложенных вам полей. Следует также указать его длительность и определить статус: неактивное, подлежащее наискорейшему выполнению [Do ASAP], запланированное или перепорученное. Метка «неактивный» [inactive] нужна для тех вещей, забыть которые нельзя, но можно благополучно проигнорировать в ближайшем будущем, пока вы не решите убрать эту метку. Однако если вы припишете неактивному действию стартовую дату, она автоматически возникнет в *ThinkingRock*, когда назначенный день наступит.

Признаем, что испытываем смешанные чувства насчет *ThinkingRock*. Если хотите буквально воплотить ДВП в Linux, эта программа для вас. В то же время мы опасаемся, что его работа и интерфейс покажутся некоторым пользователям слишком негибкими и чрезмерно опекающими. Более серьезная проблема *ThinkingRock* – частые зависания, как у *GTD-Free*. Это, опять-таки, может быть скорее проблемой Java, но так или иначе, *ThinkingRock* менее надежен, чем его конкуренты.



› Мешанина окон показывает, как точно *ThinkingRock* воспроизводит метод ДВП – тут есть все!

LINUX FORMAT Вердикт

ThinkingRock

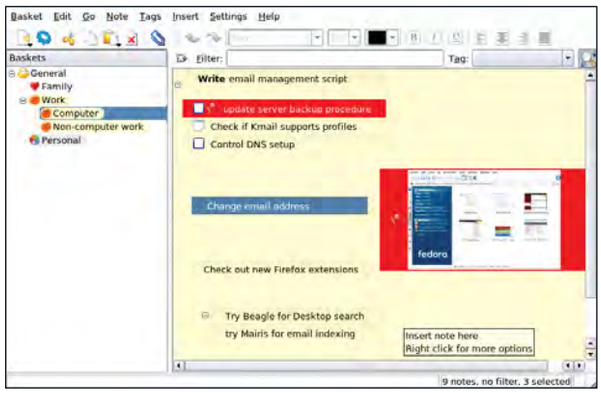
Версия: 2.0.1
Сайт: www.thinkingrock.com.au
Цена: Бесплатно

» *Нет более ДВП-шного приложения для Linux, чем ThinkingRock, но некоторые пользователи могут посчитать его непреодолимым.*

Рейтинг 7/10

Basket

Цифровая версия дверцы холодильника, в стиле KDE.



› *Basket* всегда можно использовать как расширение для Kontact, а уж там есть и календарь, и почта, и многое другое!

Basket [англ. корзина] – это контейнер... корзин, то есть мест для складирования цифровых заметок, примерно так, как заметок-стикеров на двери холодильника. В некотором смысле это удобно, так как ДВП – это свойства и корзин, и заметок.

Создавайте столько корзин и под-корзин, сколько захотите; каждой можно придать свой значок, фон и пароль. Расположение дозволяется как с фиксированными столбцами, так и «свободное», то есть можно помещать и перекрывать заметки, как вам заблагорассудится.

Заметки могут содержать почти все: текст, рисунки, ссылки на документы или на запускаемые программы. Создать заметку – проще некуда: введите что-нибудь внутри корзины, и вокруг вашего текста возникнет прямоугольник.

С каждой заметкой можно ассоциировать (крохотный!) тег-значок. Предопределенные теги в основном касаются сортировки или контроля исполнения: они имеют значения типа приоритет [priority], состояние выполнения [progress status], информация [information] или to do. У последних есть небольшой квадратик, щелчок на котором помечает заметку как «сделано» [done]. Вы также можете определить собственные теги. Если вы объедините вместе связанные друг с другом заметки, *Basket* спрячет все, кроме первой: чтобы открыть группу заново, щелкните по маленькому значку «–» на левой границе заметки.

Можете импортировать данные из других приложений типа *KNotes* или *Tomboy*, или из текстовых файлов. Корзины можно сохранять в двоичном формате или преобразовать в HTML – пока это единственный способ распечатать что-нибудь из *Basket* (должно быть исправлено в KDE 4).

В *Basket* нет календаря, проектов или контекстов, так что довольно сложно угадать в нем приложение ДВП. Тем не менее, множество хваленых бумажных реализаций ДВП являются ничем иным как пачкой карточек 3x5", налепленных на какую-нибудь стену. *Basket* – просто программа с таким же внешним видом и быстрым интерфейсом, так почему бы не дать ей шанс?

«Проектов и контекстов нет; приложение ДВП в Basket угадать сложно.»

LINUX FORMAT Вердикт

Basket

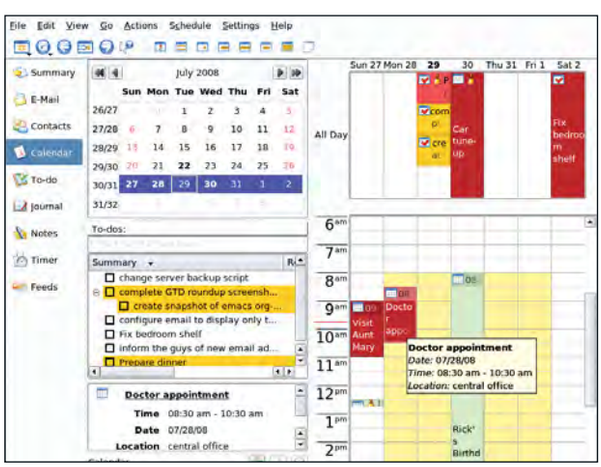
Версия: 1.0.2
Сайт: http://basket.kde.org
Цена: Бесплатно

» *Basket удобен, но не как ДВП-утилита… а как простая цифровая замена вашим карточкам 3x5".*

Рейтинг 6/10

Kontact

Оболочка в KDE для персонального менеджмента.



› Почта, to-do, заметки, календарь, таймер и еще многое: *Kontact* объединяет все это и некоторые другие функции.

Делать обзор такого приложения и просто, и сложно: строго говоря, *Kontact* – не приложение, а интерфейс личного органайзера для рабочего стола KDE – оболочка, куда можно включить *KMail*, *KOrganizer*, *KNode*, *KNotes*, *Kalendar*, *Basket* и все что угодно. Такие залежи приложений могут навести на мысль, что *Kontact* занимался ДВП задолго до рождения Дэвида Аллена, чего на самом деле нет.

Давайте бегло рассмотрим, что предлагает *Kontact*, а затем проверим, как он применяет сам себя для работы по ДВП. У вас в распоряжении есть адресная книга, два хитроумных клиента для электронной почты, календарь и заметки с менеджерами to-do-списков. Можно создавать иерархические задачи to-do с прикреплениями и с разными приоритетами, затем упорядочивать их по разным критериям или отображать в календаре. В календаре также появляются дни рождения и другие специальные даты из адресной книги. Более важно то, что полный вид *Kontact* отображает в одном окне сразу все неоконченные to-do (обогащенные статусной информацией), встречи на следующие несколько дней, все почтовые папки с непрочитанными сообщениями и – не последнее дело! – все заметки, введенные в *Kontact*.

Это ДВП? Не совсем. Это персональный органайзер со всей отделкой и технологией KDE для его поддержки; несомненно, здесь избыток того, что понадобится многим пользователям для поддержки своих ежедневных задач. Однако попытка заставить *Kontact* иметь ортодоксальный интерфейс ДВП вряд ли будет самым эффективным решением.

С этой точки зрения, кое-чего все еще не хватает: например, реальных ссылок и объединения различных объектов среди компонентов *Kontact*, таких как почта и статьи RSS (вспомните *Chandler*).

Другое препятствие к использованию *Kontact* как ДВП – отсутствие ДВП-шного общего вида. *Kalendar* хорошо работает как напоминалка, но где там списки очередных, ждущих и перепорученных действий? Нету.

LINUX FORMAT Вердикт

Kontact

Версия: 3.5.9
Сайт: http://kdepim.kde.org
Цена: Бесплатно

» *Kontact выполняет все, что нужно пользователю от персонального информационного менеджера, но ДВП в нем не органично.*

Рейтинг 7/10

Утилиты GTD

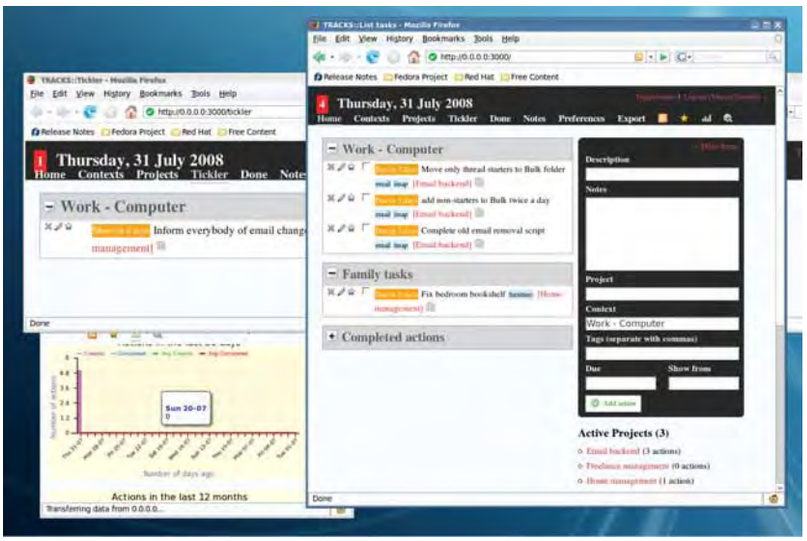
Вердикт

Tracks 8/10

Ничто в этом мире, за исключением очень талантливого человека, которому платят, чтобы он круглые сутки был вашим персональным секретарем, не сможет удерживать вашу жизнь под контролем, если вы не привыкли все время серьезно следовать правилам. Говоря о ДВП, все специалисты согласны, может быть, только в одном: если вы не будете делать регулярный, полный анализ, ДВП вам не поможет, какую бы реализацию вы ни избрали.

Другая истина – то, что работает для вас, может и не работать для других. Вам нужно отыскать систему, которая больше всего подходит образу ваших мыслей, вашей работе, вашей личной жизни и тому, как вы пользуетесь компьютером. Если вы ненавидите графические оболочки, вы уже предрасположены к тому, что Org-mode *Emacs*'а станет для вас прекрасным инструментом.

Облегчив таким образом душу, назовем победителя. Главными функциями, перечисленными нами во введении, были более или менее явное следование ДВП, быстрый интерфейс ввода данных, работающий даже при отсутствии Интернета, и стабильность. Именно поэтому мы и объявляем, что победителем стал *Tracks*. Он содержит в общем-то все и даже более из того, что мы перечислили, а заодно статистику продуктивности и множество расширений и готовых скриптов для импорта/экспорта данных во всякие форматы. Да, там нет почтового интерфейса, но *iCal* и ленты RSS прекрасно работают с наиболее популярными в Linux клиентами почты, календаря и новостей, включая *KMail* и *Evolution*. Конечно, никто не назовет чтение лент *Tracks* из этих клиентов полной интегрированной системой PIM/GTD, но кто знает? Если вам действительно нужен ДВП, такое сочетание, во всяком случае, работает! LXF



› ***Tracks*** **имеет почти все нужные функции, плюс несколько приятных дополнений.**

> «Tracks содержит все и даже более из того, что мы перечислили.»

Обратная связь

Нужны ли вам средства ДВП? Используете ли вы эти программы или же что-то совершенно другое (но, конечно, из области открытого ПО!)? Напишите ваше мнение по поводу нашего обзора на letters@linuxformat.ru.

Таблица возможностей

Название	Версия	Рабочий стол	Формат базы данных	Интеграция с почтой	Календарь	Работает с браузером	Язык
Emacs Org-mode	6.06b	Консоль	Простой текст	✓	✓	☒	Lisp
GTD-Free	0.3.1	Любой	XML	☒	Частично	☒	Java
MonkeyGTD	3.0 alpha	Любой	XML	☒	Частично	✓	Php
Tracks	1.6	нет	Sqlite 3, MySQL или PostgreSQL	☒	Частично	✓	Ruby и JavaScript
Chandler	1.0-rc1	Любой	XML	✓	✓	☒	Python
ThinkingRock	2.0.1	Любой	XML	☒	Частично	☒	Java
Basket	1.0.2	KDE	Двоичный	☒	☒	☒	C++
Kontact	3.5.9	KDE	Несколько	✓	✓	☒	C++

Distrowatch

Ежемесячная сводка новостей дистрибутивов Linux.

СЬЮЗАН ЛИНТОН является владелицей и оператором tuxmachines.org.

Право выбирать

Некая заноза в моем мозгу вновь побеспокоила меня после выхода ядра версии 2.6.25, вследствие возобновления споров о чистоте открытого кода и проприетарных включениях. Споры эти длятся почти столько же, сколько существует свободное ПО, но мне о них стало известно, когда Nvidia выпустила графические 3D-драйвера для своих карт под Linux. На тот момент владельцы Nvidia были в восторге, но очень скоро при каждой загрузке стало появляться предупреждение о том, что ядро «подпорчено» [tained].

С годами споры поутихли, но несколько месяцев назад вынужденно возобновились после выхода Linux 2.6.25, не предоставляющего доступа к символам, помеченным как «только GPL». Это сделало многие проприетарные беспроводные и графические чипы бесполезными или ущербными. Не лучший выход.

Как насчет выбора?

Я понимаю аргументы каждой из сторон, но не важнее ли право пользователя на выбор? Ограничение прав пользователей не входит в философию Open Source. Я говорю не о букве GPL, а об идеологии. Open Source – это больше, чем лицензия. Это больше чем открытый код. Это понятие включает также сообщество пользователей – многие из которых обращаются к свободному ПО, чтобы получить свободу выбора.

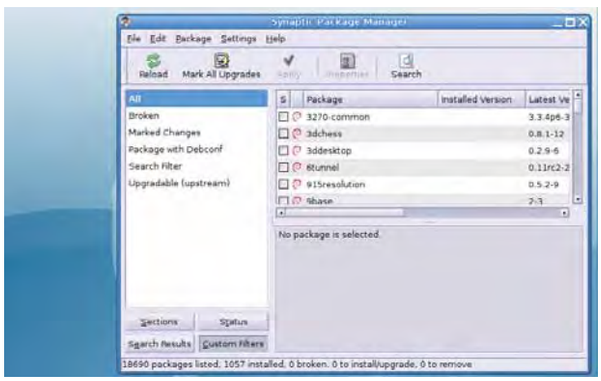
Модули, брюзжа, вернули, но споры продолжаются. Если мне требуются двоичные драйвера, то я должна иметь возможность заполучить их без порицаний или наказаний за это. Кто способен выжить на чистом открытом коде, пусть выживает, но уж гарантируйте такое же отношение тем из нас, кому не обойтись без закрытых двоичных модулей. Неужто у нас всех нет права выбора?

lxf.distrowatch@futurenet.co.uk

На полдороге

Debian Etch-с-половиной: промежуточный релиз с поддержкой новых устройств и включением Linux-2.6.24.

Debian GNU/Linux – один из наиболее уважаемых патриархов мира Linux. В этом году он отмечает свое 15-летие, а также является одним из старейших дистрибутивов Linux, все еще находящихся в разработке. Debian, вероятно, наиболее распространенный дистрибутив в сегменте серверов, но он также предоставляет популярные окружения рабочего стола, такие как KDE, Gnome и *Xfce*.



Для управления пакетами Debian использует *Synaptic* поверх *APT*.

Не будучи чересчур дружелюбным к пользователю, Debian породил множество ответвлений, направленных на заполнение данной ниши. Он послужил основой для многих популярных настольных дистрибутивов, таких как Ubuntu, Mepis, Sidux, Damn Small Linux и Parsix GNU/Linux. *APT* – система управления пакетами в Debian – стала одним из наиболее почитаемых и надежных механизмов установки, удаления и обновления ПО в Linux. Его основатель и разработчики считаются одними из наиболее влиятельных членов сообщества Open Source. Если это не впечатляющая родословная, то уж и не знаю, чего вам еще надо.

Debian GNU/Linux 4.0 был выпущен в апреле 2007 года и с тех пор обновлялся несколько раз. Ветвь разработки Debian 5.0 недавно была заморожена; в промежутке вышел Debian 4.0r4, по прозвищу Etch-and-a-half [Etch-с-половиной]. Впервые в истории Debian основной целью релиза явилась поддержка нового оборудования. Те пользователи, которые имели проблемы совместимости или поддержки в 4.0, должны найти 4.0r4 более приятным.

> «Главная цель данного релиза — поддержка новых устройств.»

Обычный метод обновления Debian – использование *APT*; однако также предоставляется установка по сети, с CD и DVD. Прилагается графический установщик – благодаря чему инсталлировать Debian не сложнее, чем другие дистрибутивы, включая разметку диска и выбор групп пакетов, а также настройку учетных записей пользователей, пароля root и загрузчика. В нем отсутствует несколько продвинутых опций, имеющихся в некоторых других установщиках, но он быстр, устойчив и относительно прост в использовании.

Достижение данного релиза – ядро 2.6.24, предоставляющее поддержку новейших устройств и улучшения драйверов, а также множество исправлений безопасности. Кроме обновлений обычных программ, введена поддержка новых видеокарт Nvidia и Intel в *X.org*, расширена совместимость с другими приложениями и улучшена работа с двумя мониторами. Усовершенствована работа *Sysvinit* с новыми SATA-накопителями, а Wireless Tools были доведены до уровня нового ядра. Поскольку это лишь Etch-с-половиной, то Gnome остался в той же версии 2.1.4, а KDE – 3.5.5, и *Iceweasel* – 2.0.0.16.

Большинство устройств обнаруживается и конфигурируется автоматически, и Debian поставляется со множеством инструментов для настройки, например, принтеров, даты и времени, сети и учетных записей пользователей. Оформление фирменное и притягательное. Однако наиболее веский довод за использование Debian – его приверженность к качеству, защищенности и стабильности. Что весьма очевидно в этом последнем релизе. Если вы ждете 5.0 – терпя ужасную заторможенность цикла разработки Debian – вероятно, Etch-с-половиной может удовлетворить ваши нужды сейчас.

www.debian.org

Жажда скорости

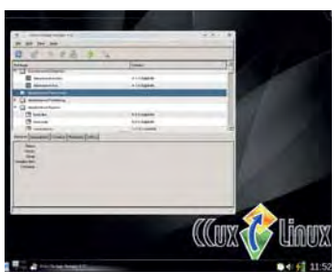
CCux 0.9.9 CCux укомплектован KDE 4.1, Linux-2.6.26 и молниеносным быстродействием.

Более чем два года спустя, CCux Linux вернулся лучшим, чем когда-либо, в версии 0.9.9. CCux Linux – это дистрибутив, оптимизированный по скорости путем компиляции с нуля для i686.

CCux – большой DVD для загрузки, предлагающий вам богатство выбора. В процессе дружелюбной установки можно указать один из нескольких рабочих столов (включая KDE 4.1), а также офисные пакеты, мультимедиа-проигрыватели, приложения для Интернета и многое другое. Процесс установки упрощается опцией «полная инсталляция», которая приведет к заполнению меню CCux полезными и забавными приложениями.

Например, среди офисного ПО – *AbiWord*, KOffice, *OpenOffice.org* и *Scribus*. В мультимедиа-проигрывателях представлены *MPlayer*, *Dragon*, *Video Lan Client* и *Xine*. Внутри – Linux-2.6.26, *X.org 7.3.0* и *GCC 4.3.1*. *Smart Package Manager* выполняет добавление, удаление и обновление приложений после установки.

Кроме великолепной линейки приложений, настоящей звездой CCux является производительность. Быстрая загрузка – это лишь начало.



› ***Smart Package Manager*** **устанавливает приложения из RPM-пакетов.**

Приложения открываются и работают быстрее, меню появляется мгновенно, и нет признаков артефактов графики.

Конечно же, ничто не идеально. Продвинутся система энергосбережения кажется несколько ограниченной на некоторых ноутбуках, ввиду отсутствия их поддержки в ядре. Некоторые сторонние пакеты с исходными текстами не собираются. Кроме того, часть ваших любимых приложений может отсутствовать в репозиториях CCux. Но, подытожив все плюсы и минусы, CCux – определенно интересный дистрибутив, заслуживающий места на вашем жестком диске.

http://ccux-linux.de

Музыку нам в ухо

Musix GNU+Linux Исчерпывающий пакет для музыкантов внутри нас.

Musix GNU+Linux – мультимедийный дистрибутив, цель которого – предоставить приложения для создания музыки, видео и графики; при этом он содержит основные инструменты для офиса и Интернета. Он полностью основан на Knoppix и поставляется как установочный LiveCD на нескольких языках, таких как испанский, английский, немецкий, французский и итальянский.

Вследствие Knoppix-основы, доступны все те же загрузочные опции для настройки различных устройств и выбор языка. При входе предлагаются рабочие столы *IceWM*, *Fluxbox* и KDE, и меню набиты приложениями.

От *AbiWord* до *ZynAddSubFX*, вы найдете все, что понадобится вам для набора текстов, публикации и печати партитур, настройки гитары, практики в теории музыки, создания и редактирования 3D-графики и видео, удаления шумов, добавления спецэффектов, путешествия в Интернете, записи музыки, общения с друзьями и тому подобного. Linux 2.6.21, *X.org 7.1.1* и *GCC 4.1.2* образуют базу для таких приложений, как *Rosegarden*, *Ardour*, *GIMP*, *Tartini* и *Xine*. Musix также включает полный набор настольных приложений на каждый день, а также инструменты настройки, мониторинга и управления пакетами в репозиториях Musix, Debian и 64 Studio.

› ***Musix Control Centre*** **содержит модули для всей системы.**

Обновленный Musix 1.0R2R5 был выпущен в августе. Ведущий разработчик, он же – учитель музыки, сообщил, что этот релиз является источником гордости и удовлетворения для его команды и что после шести месяцев тестирования серьезных ошибок не осталось. Musix – прекрасная система и для тех, кто создает мультимедиа, и для тех, кто просто наслаждается им.

www.musix.org.ar

Командная строка всегда в моде

Я все еще использую Bash в качестве файлового менеджера. В основном из-за удобства: ведь командная строка куда проще и быстрее, чем череда щелчков вдоль и поперек графического менеджера файлов. По-моему, каждый должен выучить хотя бы пару основных команд, работающих в любом дистрибутиве Linux. Предлагаю вашему вниманию несколько наиболее распространенных вариантов:

Команда	Выполняемое действие
ls	Выводит содержимое текущего каталога.
cat	Выводит содержимое файла в stdout (окно терминала).
cd	Меняет каталог или на основе полного пути, или на любой подкаталог текущей папки.
cp	Копирует файлы в каталог.
su	Сменить пользователя (обычно на root – администратора).
rm	Удалить файлы или каталоги.
mv	Переместить или переименовать файлы или каталоги.
df	Вывести список и основную информацию о всех примонтированных файловых системах.
mount	Примонтировать раздел или сменный диск.
mkdir	Создать новый каталог.

Хит-парад дистрибутивов

10 самых посещаемых страниц на DistroWatch.com со 2 июля по 26 августа 2008 (визитов в среднем за день)

Дистрибутив	Число визитов	
1 Ubuntu	1,825	‹›
2 OpenSUSE	1,512	‹›
3 Mint	1,196	‹›
4 Fedora	1,152	↑
5 PCLinuxOS	1,056	↓
6 Debian	1,026	‹›
7 Mandriva	962	↑
8 GOS	633	↑
9 Damn Small Linux	555	↑
10 Slackware	513	↑

» **DistroWatch.com** оценивает популярность дистрибутивов, основываясь на числе визитов на страницу каждого из них. Хотя это не дает представление о действительном числе установок, но показывает, какие дистрибутивы были более популярны за определенный промежуток времени. lxf

Фараон

Карл Ворт потратил шесть лет жизни на создание Cairo, инструментария векторной графики, и считает, что самое интересное только начинается…

Всего два года назад в Лионе, Франция, на первой конференции Libre Graphics Meeting (LGM) Карл Ворт [Carl Worth] пытался заинтересовать народ инструментарием векторной графики под названием Cairo — он был создан исключительно для рисования, и больше ни для чего. Его принялись портировать в операционные системы, в том числе OS/2 и BeOS, и на сегодня Cairo является основной графической библиотекой многих приложений. LXF состыковался с Вортом, чтобы докопаться до самой сути: как выглядит Cairo — и ваш рабочий стол — и каковы его виды на будущее…

Linux Format: Ни в KDE, ни в Gnome не найдешь пункта меню для Cairo, поэтому расскажите, пожалуйста, что он, собственно, делает?

Карл Ворт: Создает графику. Он – кроссплатформенный и решает многие задачи, например, выводит изображения на дисплей, которым может быть *X* или *Quartz* на Mac OS X, и для этого не понадобится ничего менять в коде вашего приложения, кроме единственной строки, создающей поверхность. Cairo выводит результат в PDF или SVG, или же в файл изображения. По-моему, одна из лучших функций, которые Cairo предлагает приложениям – это унифицированный API, способный и отображать, и печатать. Цель – суметь изобразить все, что нужно для двумерного приложения.

LXF: Разве самая последняя работа не велась над PostScript и PDF?

KB: Главное различие заключается в том, что мы делали с выводом в PostScript и PDF. Мы говорили, что результат в PDF должен всегда выглядеть ничуть не хуже того, что отображается на экране. Раньше нам удавалось добиться этого просто превращением в растровое изображение с высоким разрешением и выводом результата в PostScript… не очень удобно.

В 1.4 у нас был код, который сам по себе мог многое, но были и некоторые погрешности, исправить которые никак не удавалось: если, например, вы рисовали градиент, который легко выразить средствами PostScript, он все же становился растровым. Все подобные проблемы мы исправили в версии 1.6. Теперь результат выводится в PostScript и PDF совершенно нормально и удобно. Я могу этим хвастаться, потому что не писал ни строчки данного кода – все сделал Эдриен Джонсон [Adrian Johnson], фантастический член сообщества, который возник из ниоткуда и потратил на эту работу фантастическое количество часов. Впервые нашим выводом в PDF можно пользоваться.

LXF: А что было самым трудным?

KB: Встраивание шрифтов. Должен сказать, с этим у нас было множество непонятных ошибок. Вроде делаешь все правильно, то есть и вправду правильно, согласно спецификации PostScript –

графики

но стоит отправить все это на печать, как принтеры начинают мигать и просить, чтобы их перезагрузили, потому что им отправили файл PS. Мы реализовали то, что говорится в спецификации, но не совсем так, как это делается у всех остальных.

LXF: И как в это втянулся Эдриен?

KB: Изначально Эдриен занялся реализацией поддержки CID-шрифтов, ну, и прочими подобными вещами. Я думал, он работает в какой-то компании, использующей Cairo со своим приложением, которое гребет им миллионы долларов. А встретился я с ним в Австралии в январе, и он мне сказал: «О нет, я всем этим занимаюсь в свободное время». Я спросил: «И что ты делаешь?» А он ответил: «Я столько раз генерировал файл PDF, и он оказывался кривым, все перепутано: шрифты не те, верстка не та… Меня это просто достало. Я хочу, чтобы все работало нормально, и я слышал о Cairo. Я подумал, что если он будет создавать PDF, то все станут им пользоваться, а я позабочусь, чтобы он работал правильно».

LXF: И насколько вы близки к своей цели – чтобы все перешли на Cairo? Реально ли для Cairo стать тем Единственным, что правит всеми? [в оригинале *the one to rule them all* – намек на «Властелина Колец» Р.Р.Толкиена, – прим. пер.]

KB: Для меня было отчасти сюрпризом то, что я увидел на конференции Libre Graphics: помните эти пять логотипов – так вот, три из них в разной степени используют Cairo; я такого не ожидал, потому что пару лет назад на первой LGM им не пользовался никто. У нас уже есть *GDK*, у нас есть Gnome, который применяет его для печати, у нас есть Mozilla, *Scribus*, *Inkscape*, *OpenOffice.org* и *GIMP* – все хоть как-то, да пользуются им. Некоторые – совсем чуть-чуть, где-нибудь в уголочке, но и это уже здорово, это – отлично.

LXF: А что, по вашему мнению, является катализатором, заставляющим людей выбирать Cairo? Все-таки переход на него требует известных усилий.

KB: *GDK* начал использовать Cairo в версии 2.8. Конечно, он сильно отличается от всех прочих: все прочие – приложения, а *GDK* – библиотека, и в частности, она не оборачивает API Cairo, она передает его напрямую приложениям. Одной из целей работы над Cairo было создание действительно простого в использовании API. Мы потратили почти два года, улучшая API так и этак, прежде чем выпустить релиз 1.0 – старались сделать его как можно проще в использовании. Я всегда говорю людям: если вам не в кайф работать с Cairo, значит, что-то неправильно. Мы решили, что взаимодействие с ним должно доставлять удовольствие, и я надеюсь, нам удалось этого добиться. Спросите любого разработчика *GDK*, и он вам скажет: «просто используйте Cairo».

LXF: А *Firefox* тоже использует Cairo?

KB: Сейчас *Firefox 3* применяет Cairo для отображения всего своего контента: и основной панели, и виджетов. Все это нарисовано в Cairo, на всех платформах. Раньше в Mozilla было три или четыре отдельных части кода, одна – для рендеринга под Linux, одна – под OS X (Cocoa), одна – под Windows, и еще одна – чтобы генерировать результат в PostScript; и они смогли убрать все эти части и заменить их на один кусок кода, создающий графику с помощью Cairo. Им немножко не повезло, поскольку они были первопроходцами: выяснилось, что наши движки под OS X и Windows работают похуже, чем остальные, и в итоге в Mozilla проделали колоссальную работу, чтобы привести их все в соответствие.

LXF: А много ли еще осталось такого, что Cairo мог бы улучшить?

KB: Мы заметно усовершенствовали Cairo 1.6 и вывод результата в PDF, но осталось несколько недоделок. Одна из них – многократное использование изображений в документе, например, водяных знаков. Вот на что надо обратить внимание. Кроме того, я как раз зафиксировал в коде 40 крупных изменений, которые вводят совершенно новое явление: пользовательские шрифты. Они позволят приложению предоставить контуры своих глифов, и они будут встроены в кэш глифов Cairo. То есть можно воспользоваться всеми преимуществами системы кэширования глифов, но с собственными настраиваемыми шрифтами.

Вещь, которой мы скоро займемся… Извлечение текста из PDF-файлов – идея тут такая: после того, как вы что-то напечатали, у вас должна быть возможность скопировать и вставить и снова получить текст назад.

ПРО БУДУЩЕЕ CAIRO:
«Нужно еще много чего сделать, но мы пока не знаем, как».

LXF: Чего вы ожидали от посещения LGM?

KB: Всегда отлично получать обратную связь от пользователей твоего кода. Мы еще много чего хотели бы видеть в Cairo, но пока не знаем, как это сделать. И потом, сейчас в Cairo есть только цветовая спецификация RGB. Вы можете добиться чего-то еще, использовав непрозрачный цвет, добавив RGB-контрольную точку или передав нам буфер RGB-данных, но у нас отсутствует возможность связи со CMYK или любым другим цветовым пространством.

Нам также очень нужны фильтры – в первую очередь, фильтр размывки изображения. Думаю, что было бы также неплохо взять часть GASL и передать ее Cairo, и если ваша цель – X или что-то в том же духе, это все передавалось бы на видеокарту и запускалось, а если ваша цель – PDF, то компилировалось бы на CPU, запускалось, потом бы вы получали нужный растровый результат и передавали в PDF-файл. Всем нужна высокая точность, но нужна также и выразительность, особенно для аппаратного ускорения… Каждый хочет уметь говорить на разных языках; по-моему, это было бы действительно здорово. LXF

» Рубрику ведет
Евгений Балдин

» **Кто**
Сергей Геннадьевич Бессонов, к.т.н., директор ООО «Элсис». Закончил Нижегородский государственный технический университет по специальности «Автоматизация и управление».

» **Откуда**
г. Нижний Новгород.

» **Цель**
Создать бизнес на внедрении свободного программного обеспечения.

Бизнес на внедрении

Одним из значительных препятствий на пути распространения открытого программного обеспечения является отсутствие поддержки на местах. Его ликвидацией в Нижнем Новгороде занимается ООО «Элсис» и **Сергей Бессонов**.

В сторону Linux я начал заглядываться достаточно давно – ещё в 1998 ставил Red Hat 5.2; но серьёзное использование началось только с 2000 года и Mandrake 8.0. Начиная с 2005 года, участвую в свободных проектах в качестве разработчика. В основном, готовлю патчи и исправляю ошибки в разных программах. Вылавливать недочеты в СПО – это моя страсть. Один раз мою работу даже особо отметили на Novell Bugzilla (тогда я пользовался OpenSUSE 10.1 – мне нравится менять дистрибутивы) и за количество размещённых и исправленных ошибок подарили коробочную версию OpenSUSE 10.2.

ООО «Элсис» (ЭЛектронные СИстемы Связи) изначально создавалась для совсем другого проекта. Одна из проблем доступа в Интернет в Нижнем Новгороде – это монополия компании «Волга-телеком», что позволяет более мелким фирмам вроде нашей играть на завышенных ценах. В разнице между себестоимостью и среднерыночной ценой Интернета вполне можно «уместить» несколько посредников. Идеей для организации компании было создание набора хотспотов по образцу Golden Wi-Fi в Москве, однако буквально через два месяца в город пришли первые провайдеры PreWiMAX, и мы отказались от этого проекта как от бесперспективного. Пришлось искать новые направления работы. Так как к этому переломному моменту у нас собрались ведущие Linux-специалисты города, то избранным направлением стало оказание предприятиям услуг, связанных с GNU/Linux и свободным программным обеспечением. Теперь даже название компании расшифровывается по-другому: Elsys – это Enterprise Linux SYStems. В нашей

компании работает семь человек, и среди них не только специалисты по внедрению, но и преподаватели, PR-менеджеры.

На начальном этапе разработки нового для нас направления, когда у нас не было сформированных решений, мы просто предлагали услуги «знающих людей» и изучали потребности малого и среднего бизнеса, стараясь по возможности их удовлетворять. Сразу скажу, что не всё получалось. Затем у нас появились готовые решения, которые мы могли уверенно предлагать как надёжное «средство от проблем» с примерами внедрения; они значительно более дёшевы, чем лицензирование имеющегося у фирм ПО. Не секрет, что большинство клиентов – это либо те, кто ещё не начал бизнес, но уже ужаснулся от цен на продукты, которые необходимо приобрести, либо те, кто до сих пор сидит на нелегальном закрытом ПО и не может «выйти из тени» по той же причине. Т.е. наши клиенты преимущественно хотят сэкономить, хотя есть и такие, для которых использование СПО – это способ расширить функционал применяющихся решений.

Основной профиль нашей деятельности – это автоматизация бизнеса. Здесь, кроме 1С, обязательно используется *OpenOffice.org*, *Firefox*, *Thunderbird*, иногда *PostgreSQL*, *Samba*; однако часто клиентам бывает необходимо и индивидуальное решение задач. Клиенты вообще любят индивидуальный подход, и типовыми решениями ограничиться не получается. Мы также внедряем такие программы, как *Zimbra Collaboration Server*, *ZoneMinder*, *Zenoss*, *Squid*. Разумеется, многим требуется web-сервер на базе стека LAMP.

Пока далеко не все программы (взять ту же 1С) удаётся заменить свободными аналогами без существенных издержек. Способ совмещения, казалось бы, несовместимого, пришёл, как обычно, через личный опыт. Когда я ещё писал диссертацию, то подумал, что совместимости ради неплохо бы поставить *Microsoft Office 97* на своем компьютере (так сложилось, что эта версия офиса использовалась у нас на кафедре). Виртуальная машина по тем временам была не слишком удачным решением, так как памяти для комфортной работы не хватало, а простой *Wine* ещё не позволял запускать *MSO*. Одним из вариантов был *CrossOver Office*, но я нашёл для себя выход получше: за 400 рублей купил Wine@Etersoft Local (это моя единственная покупка, так как *Microsoft Office* мне в своё время подарили) и запустил *MSO* на нём. Помню, техподдержка очень удивилась, узнав, что я купил Local не для запуска 1С, а для других целей. Таким образом, мосты с Etersoft были наведены, один экземпляр их программы у нас имелся, и мы могли спокойно протестировать качество его работы с 1С и другими приложениями. Ну, а потом мы связались с фирмой Etersoft, заключили контракт дилера и стали поставлять их решения в составе наших.

Из закрытого ПО, для своих нужд мы сейчас используем только 1С для ведения налоговой отчётности, да ещё программу клиент-банка. Наши же клиенты выставляют совершенно различные требования по поводу необходимых им программ. Кому-то достаточно того же 1С, а кому-то требуются специализированные приложения, связанные с деятельностью предприятия.

Сегодня для работы у нас используется два базовых дистрибутива. Для бизнеса это Ubuntu, а для школ и государственных организаций – ALT Linux. Такой выбор сложился исключительно исторически. В случае с Ubuntu это связано с достаточно широким распространением данного дистрибутива в России и в мире: соответственно, для него несложно найти «эникейщика». Это ценится в коммерческих организациях. Что касается ALT Linux, то мы внедряем его в образовательных и государственных учреждениях по той причине, что компания предоставляет нам методики и поддержку для такого внедрения.

Следует отметить, внедрение Linux только-только начинает быть прибыльным: ещё года три назад мы бы просто не нашли клиентов, а без них любой бизнес обречён. Проще и выгоднее оказывать поддержку, и если есть возможность создать именно консалтинговую фирму, то лучше это и сделать. Но чтобы поддерживать, надо сначала внедрить, поэтому от внедрения, как от первого шага, никуда не деться. В свою очередь, чтобы внедрять, нужны решения, нужны подходы, нужно готовое портфолио. Старт в этом бизнесе не слишком прост.

Сейчас у нас есть вполне определённые планы на будущее. Мы не собираемся ограничиваться той инфраструктурой, что у нас уже имеется, и будем изучать новое ПО и планировать его к внедрению. Также будем активно участвовать во внедрении Linux в образовательных учреждениях и создавать курсы по обучению Linux. Будем, так сказать, сами выращивать себе клиентов.

Хочется подчеркнуть и роль сообщества. У нас постоянно возникали вопросы «как лучше сделать», «чем решить ту или иную задачу», «какое ПО лучше всего подходит для…». Ответы на них оптимально получать от тех, кто уже опробовал эти решения в реальности. Мы тоже в долгу не остаёмся: наши специалисты постоянно участвуют в форумах и списках рассылки и в деятельности нашего нижегородского LUG (http://lug.nnov.ru/). Это, кстати, и нам пользу приносит, так как активное участие в жизни сообщества не позволяет терять квалификацию. LXF

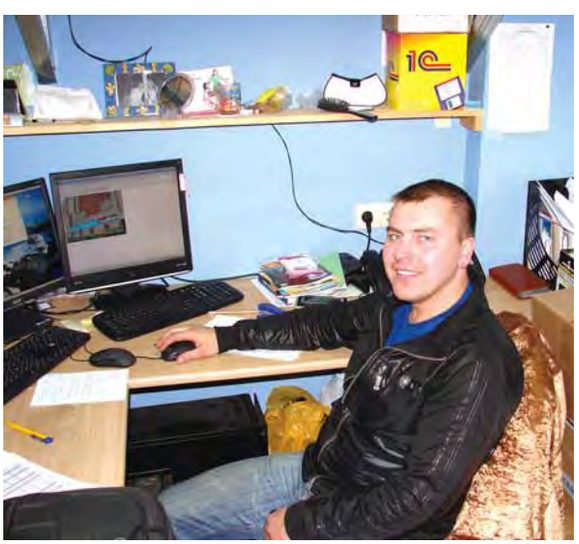
Мы продолжим разговор о внедрении Linux в Нижнем Новгороде на **с. 102**

Слово клиенту: Коротких Андрей Геннадьевич

Наша фирма занимается продажей автозапчастей и автоаксессуаров. Компьютеры в нашей работе играют одну из ключевых ролей, так как на них завязан весь процесс торговли. У нас пять компьютеров, с которых мы осуществляем приём и учёт товаров, обеспечиваем продажи, выходим в Интернет для мониторинга цен, поддерживаем работу бухгалтерии. В общем-то, на них действительно в буквальном смысле стоит весь наш бизнес.

Основной причиной для перехода на открытое программное обеспечение было желание легализоваться, затратив при этом минимальные средства. Кроме того, мы бы хотели сделать так, чтобы вирусы не останавливали работу магазинов.

На компанию «Элсис» мы вышли через моего товарища, с которым работали вместе. Жаль тех людей, которые не знают о существовании этой фирмы и такого привлекательного решения! Нам удалось сэкономить не менее 50 тысяч рублей. Остался только один компьютер, который не переведён на Linux. Мы используем его для работы со специфическими программами, такими как клиент-банк и интерактивные каталоги запчастей, которые не удалось запустить в Linux. Из оборудования так и не заработал копир Canon. А так – в принципе, нас всё устраивает.



Наш ответ: MythTV!

Как приложение для домашнего мультимедиа, MythTV вне всякой конкуренции; правда, его настройка – сущий кошмар. У **Грэма Моррисона** есть ответ и на это.

MythTV – один из важнейших Linux-проектов на планете. Это весьма амбициозный набор приложений, задуманный как самое сердце вашего домашнего центра развлечений. Он записывает, приостанавливает и прокручивает назад телепередачи, воспроизводит музыку и видео, организует вашу коллекцию фото и DVD, просматривает Интернет, делает телефонные звонки, доставляет новости и погоду и играет в игры – и все это благодаря мощи Linux.

Еще недавно заставить *MythTV* работать было не так-то просто. К счастью, дистрибутив под названием Mythbuntu изменил ситуацию. Mythbuntu – ремикс Ubuntu, разработанный как платформа для *MythTV*, и свое дело он делает великолепно. Он включает все стандартные пакеты *MythTV*, заранее настроенные и готовые к работе, наряду с присущей Ubuntu фантастической совместимостью с оборудованием, специализированным сценарием установки и программой для управления *MythTV*. При правильном оборудовании ваша система придет в полную готовность менее чем за час, и всю оставшуюся жизнь вы будете счастливо наслаждаться невероятными возможностями, предлагаемыми *MythTV*.

> «Mythbuntu – ремикс Ubuntu, задуманный как платформа для MythTV.»

По части оборудования, единственное, чем следует озаботиться – это выбор устройства ТВ-захвата. Во многих городах доступен наземный цифровой сигнал (технология, также известная как Freeview или DVB-T). Владельцы спутниковой антенны смогут принимать бесплатное цифровое телевидение со спутника. Недавно запущенная в Великобритании служба Freesat охватывает большую часть крупных каналов, а также многие другие. Они ловятся оборудованием DVB-S, и в Сравнении устройств DVB LXF106 мы рассмотрели их основные примеры. Для целей данной статьи мы остановились на USB-брелке Elgato EyeTV – он принимает Freeview в UK; его можно просто вставить, и он заработает. Вы даже можете обдумать установку более одного устройства, если захотите записывать одновременно два канала.

А еще вам потребуется скромный ПК, способный хорошо сработаться с Linux. За мощностью не гонитесь: 1-ГГц CPU с 512 МБ оперативной памяти более чем достаточно для записи одного канала при одновременном просмотре другого. Видеокарта от Nvidia будет тоже кстати: в них обычно встроено телевизионное соединение, а драйверы работают с *MythTV* на редкость хорошо. Если у вас есть все, что нужно, читайте дальше! Вы узнаете, как собрать идеальный мультимедиа-центр под Linux.

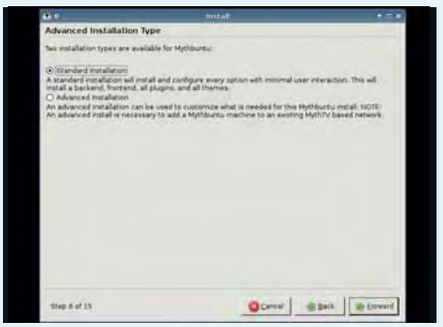
Шаг за шагом: Установка MythTV

1 Загрузитесь с диска

Первым, что вы увидите, загрузившись с **LXFDVD**, будет окно Ubuntu для выбора языка. Просто укажите свой, и выберите Установить Mythbuntu из меню загрузки. Первая половина инсталляции Mythbuntu проходит как в стандартном Ubuntu, и если вы ставили его раньше, все задаваемые вопросы будут вам вполне знакомы.

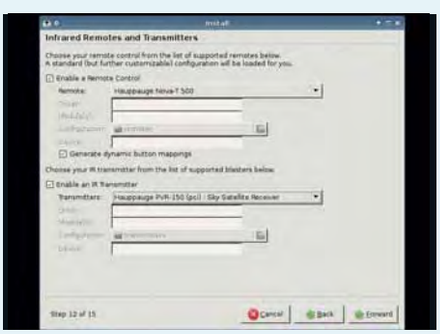
2 Выберите язык

На следующей странице вам предоставят еще один шанс выбора языка – просто нажмите Далее. Как и в установщике Ubuntu, на следующей странице вам придется как-нибудь указать ваше местоположение и продраться через ужасное обновление экрана. Управившись с этим, нажмите Далее [Forward].



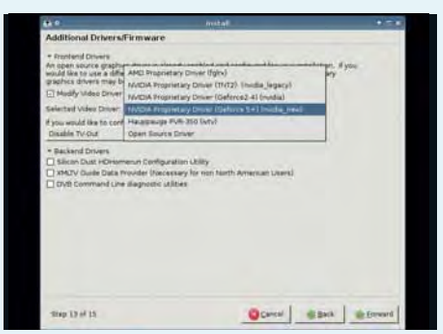
3 Стандартная установка

После указания раскладки клавиатуры и разбиения диска на разделы (естественно, следует заранее побеспокоиться о резервировании данных), вас попросят ввести имя пользователя и пароль. Затем возникнет первая страница, которая присуща именно Mythbuntu. Там дается выбор между стандартной и экспертной установками. Выберите первую и нажмите Далее.



4 Пульт управления

Вам предложат настроить пульт ДУ. Если клавиатура работает хорошо, то вам он и не понадобится, но если с ТВ-картой поставляется совместимый пульт, выберите его из выпадающего списка. Трансмиттеры не столь распространены. Они используются для смены канала на внешней телеприставке – например, на кабельном или спутниковом приемнике.

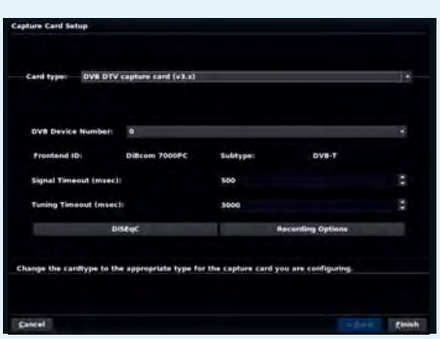


5 Проприетарные драйверы

На следующей странице настраиваются проприетарные драйверы вашего компьютера, самыми важными из которых являются драйверы видеокарты. На настоящий момент единственным жизнеспособным вариантом является Nvidia. Ее проприетарные драйверы работают очень хорошо, и большая часть таких видеокарт имеет ТВ-выход. Вам нужно выбрать подходящий драйвер Nvidia из выпадающего списка и задействовать ТВ-выход, если вы его используете.

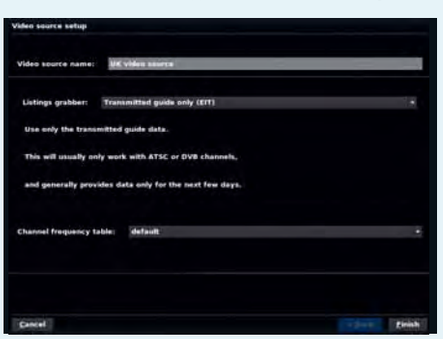
6 Настройка MythTV

Осмотрев обзорную страницу, где приведены все указанные вами основные опции, нажмите Далее, чтобы установить систему Mythbuntu, перед тем, как настроить серверную часть MythTV. На странице Общие [General] укажите формат ТВ, выбрав PAL для европейских стран или NTSC для США.



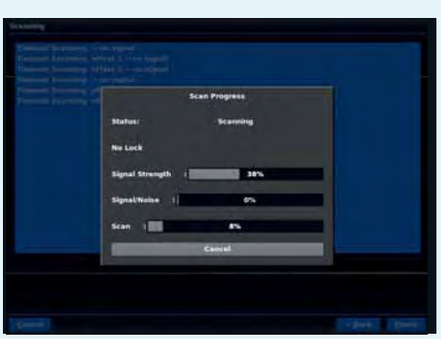
7 Карты захвата

Перейдите на страницу Карт захвата [Capture cards]. Выберите Новая карта захвата [New capture card], и в поле Тип карты [Card type] укажите DVB DTV capture card (v3.x). Этот вариант подходит для любого DVB-устройства из рассмотренных нами в Сравнении и наиболее распространен в цифровом телевидении Великобритании.



8 Источники видео

Переключитесь на страницу Источники видео. Нажмите на Новый источник видео [New video source] и добавьте для него имя. У нас им будет «UK video source». В расположенном ниже выпадающем меню Получение списков [Listings grabber] выберите Transmitted guide only (EIT), чтобы получить данные списков, затем нажмите Завершить.



9 Ввод подключений

Перейдите на страницу Подключения [Input Connections], где надо выбрать ваш ТВ-приемник и ваш Источник видео [Video source]. Щелкните по Сканировать каналы [Scan for channels], чтобы добавить каналы в базу данных, и по окончании процесса закройте приложение, чтобы запустить **mythfilldatabase**.

Слушаем музыку

Ну вот, система установлена и готова к работе; как насчет музыкальной релаксации?

Mythbuntu – идеальная платформа для прослушивания аудио: гибкая, совместимая с распространенными музыкальными форматами и готовая к запуску после установки по умолчанию. Многие тратят немало усилий на подключение медиацентра к приличным колонкам для воспроизведения теле- или видеосигнала, что имеет прекрасный побочный эффект: ваш медиацентр превращается в прекрасное средство для прослушивания музыки. Те же колонки будут замечательно звучать, даже когда вы ничего не смотрите. Удовлетворены будут и аудиофилы: многие современные компьютеры имеют порт цифрового выхода, который может быть подключен непосредственно к усилителю системы объемного звука для получения кристально чистого звучания.

Импорт аудио

Первым делом надо загнать в машину музыку. Если у вас уже есть коллекция цифровой музыки, просто перенесите ее на компьютер, являющийся медиацентром. Mythbuntu считает, что музыку держат в каталоге **/var/lib/mythtv/music**, но если это вам не созвучно, можете поменять расположение в Утилиты/Настройка [Utilities/Setup] > Настройка [Setup] > Настройки мультимедиа [Media Settings] > Настройка музыки [Music Settings] > Общие настройки [General Settings]. Верхнее поле на последней странице задает каталог. Если вы добавили музыку таким способом, то Mythbuntu потребуется заново просканировать каталог с музыкой, создавая базу данных вашей фонотеки, и это осуществляется из меню Утилиты [Utilities] > Музыкальные приложения [Music Tools] > Искать новую музыку [Scan For New Music].

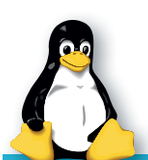
> «Многие компьютеры имеют порт цифрового выхода, ради кристально чистого звука.»

› Mythbuntu автоматически берет информацию о дорожке и исполнителе из Интернета, но если ваша мультимедийная машина не подключена к сети, вы также сможете ввести все сведения сами.

Можно также оцифровать ваши музыкальные CD. Вставьте диск в оптический привод; в главном меню выберите Оптические диски [Optical Discs] > Импорт CD [Import CD]. Ваш компакт-диск отсканируется, и в главном окне появится информация об исполнителе и о дорожках. Прежде чем нажимать на «Rip CD» для копирования содержимого на жесткий диск, рекомендуем поменять параметр качества на «среднее» [medium]. Пока Mythbuntu выполняет сжатие, откроется большое окно состояния. По завершении очередной дорожки она будет сохранена в звуковом приложении Mythbuntu, готовая для воспроизведения. По умолчанию Mythbuntu нарезает звуковые файлы в формате Ogg Vorbis, но это можно поменять в том же окне Общих настроек, которое мы уже открывали.

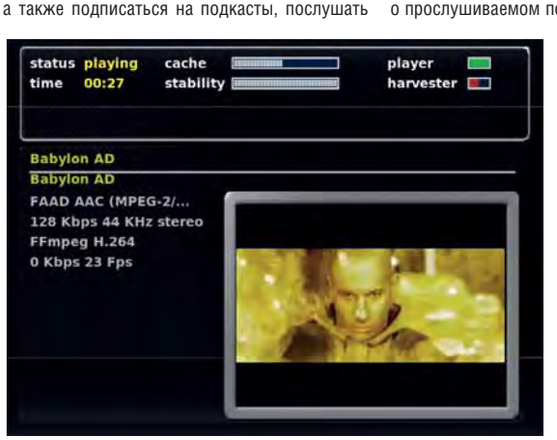
Время прослушивать

Экран воспроизведения музыки можно вызвать выбором Библиотека мультимедиа [Media Library] > Послушать музыку [Listen to Music] в главном меню. Появляющееся окно является активной очередью воспроизведения, там отображаются все дорожки, добавленные вами в список – чего мы пока не делали. Нажмите 3 или щелкните на Редактировать плейлист [Edit Playlist] для вызова редактора плейлиста. В нем можно просматривать музыкальные файлы на вашем жестком диске и выбирать каталоги или альбомы для добавления в список. Выделив папку, альбом или исполнителя, нажмите пробел; треки будут добавлены. Слева от выбранного появится галочка, а нажатие Escape вернет вас обратно в плейер. Отсюда вы можете воспроизводить дорожки из списка, перематывать вперед и назад, а также активировать полноэкранную аудиовизуализацию, включая ту, что показывает изображение обложки, которую Mythbuntu ищет в каталоге с музыкой.



Потоковое мультимедиа

Одно из лучших качеств интерфейса Mythbuntu – его безупречная интеграция потокового мультимедиа. Просто выберите Проигрывать онлайн-потоки [Play Online Streams] из Библиотеки мультимедиа, и вы увидите главное окно. Отсюда вы сможете выбрать, какие потоки смотреть или слушать, а также подписаться на подкасты, послушать интернет-радио, посмотреть онлайн-телевидение. Используйте ряд нижних кнопок для переключения между главной группой потоков, а если хотите добавить собственный, введите его URL, нажав на E, введя данные и щелкнув на кнопку Добавить новый [Add New]. Если вам нужна более подробная информация о прослушиваемом потоке, просто нажмите I.



› Чтобы насладиться прекрасным выбором трейлеров от Apple, не обязательно иметь OS X. Mythbuntu сольет их из Интернета и прокрутит на вашем телевизоре с одного щелчка.

Смотрим видео

Тревор Хорн [Trevor Horn] и Buggles заявляли, что видео загубило звезду радио, но MythTV извлекает лучшее из этих двух миров.

Mythbuntu располагает собственной функцией воспроизведения DVD, и она работает без всякой дополнительной настройки, включая меню навигации. Если у вас есть декодер внешней системы объемного звука с оптическим подключением из вашего звукового оборудования, на третьем экране страницы Утилиты/Настройка > Настройка > Общие [General] (помеченном Аудио [Audio]) вам нужно выбрать 'Enable AC3/DTS to SPDIF'. Звуковые данные будут передаваться вашему декодеру в неизменном виде, для полного сходства с кинотеатром.

В зависимости от законности создания личных копий в вашей стране, перекодирование коллекции DVD на ваш жесткий диск также является удобным способом превращения вашего медиакомпьютера в видеоавтомат. С видео, как можно ожидать, работают примерно так же, как и с музыкой. Mythbuntu поддерживает базу данных файлов всех совместимых форматов, на которые он наткнулся, и при желании что-нибудь посмотреть обратитесь к этому списку через меню Библиотека мультимедиа.

› Информация о фильмах и видео берется из Internet Movie Database с помощью простого скрипта.

Место, где хранится ваше видео, указывается на странице Утилиты/Настройка > Настройка > Настройки мультимедиа > Настройки видео > Общие настройки. По умолчанию это **/var/lib/mythtv/videos**. На странице Настройки плейера [Player Settings] также можно указать команду, используемую для проигрывания AVI-файлов. Например, переключение между *MPlayer* и *Xine* – хороший способ обойти проблемы с совместимостью у *MPlayer*. На медленных машинах можно поперебирать опции командной строки *MPlayer'a*, чтобы активизировать аппаратное ускорение видео.

Если вы ничего не поняли – не волнуйтесь: варианты по умолчанию обеспечивают широкую совместимость и неплохую производительность. Как и в случае с фонотекой, закончив добавление файлов, пересоздайте базу данных вручную; это делается выбором пункта Менеджер видео [Video Manager] из меню Утилиты/Настройка.

Поиск в IMDB

На странице Менеджер видео можно производить и другие полезные действия: например, поиск информации о фильме в Internet Movie Database (IMDB). Выберите видео, которое вы хотите снабдить информацией, и нажмите I, а затем выберите пункт Правка метаданных [Edit Metadata]. Информация, представленная на этой странице, используется для поиска в IMDB добавочной информации о видео. Если имя файла достаточно наглядно, не нужно ничего менять. Однако для лучших результатов, введите название видео и нажмите Готово [Done]. Теперь снова нажмите I, и на этот раз выберите Поиск [Search]. Mythbuntu будет искать ваше видео в IMDB и автоматически заполнять детали. Если доступна обложка, она будет скачана для использования в качестве пиктограммы фильма.

Теперь, просматривая свою коллекцию, вы увидите обложку, а также краткий сюжет и такую информацию, как год и рейтинг IMDB – это здорово для киноманов. Способ представления видео можно менять, нажимая M и выбирая между табличным и списочным представлениями. Нажатие Enter на любом из выбранных фильмов запустит просмотр. Нажатие Escape во время просмотра вернет вас в интерфейс Mythbuntu.

Для ТВ

Пульт управления
Для воспроизведения AVI-файлов Mythbuntu использует внешнее приложение под названием *MPlayer*. Это значит, что пока вы не создадите профиль пульта управления для *MPlayer*, он не будет работать при просмотре этих видеофайлов. За подробной информацией обратитесь на **www.mplayerhq.hu/DOCS/HTML/en/control.html**.

Фотографии

Как и в случае с музыкой и фильмами, ваша цифровая фотоколлекция должна лежать в каталоге **/var/lib/mythtv/** в папке под названием **pictures**. Вам не нужно грузить Mythbuntu деталями: вы увидите свою фотоколлекцию, переключившись в вид Галерея изображений [Image Gallery] из меню Библиотека мультимедиа [Media Library]. С помощью клавиши Enter можно заходить в подкаталоги и просматривать каждый рисунок в полный экран; перемещайтесь по коллекции, нажимая клавиши стрелок влево и вправо. Нажатие M позволяет зайти в меню слева, где вы можете запустить слайд-шоу и поменять настройки фото. Если у вас есть приличное оборудование для 3D-графики, задействуйте OpenGL-переходы [OpenGL transitions] на второй странице: это дает гладкие и анимированные переходы между фотографиями в режиме слайд-шоу. Вставив цифровую камеру или USB-брелок, вы сможете импортировать изображения прямо в вашу коллекцию, а также снабжать фотографии в коллекции тэгами, чтобы в кипе отпускных снимков проще было отыскать мало-мальски стоящие.

› Если у вас есть видеокарта с ускорением OpenGL, вы можете включить гладкие переходы в панели Настройка фотографий [Photos Configuration], превратив слайд-шоу в сногсшибательное трехмерное зрелище.

Просмотр и запись ТВ

Вот что Mythbuntu делает лучше всего: запись телепередач, чтобы вы могли смотреть и удалять их на досуге.

Для ТВ

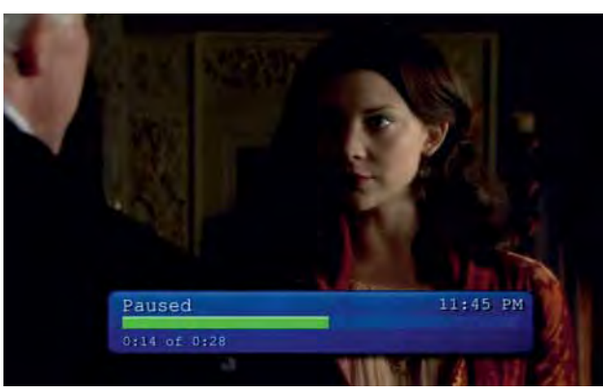
Правила записи
Для вызова редактора правил нажмите R на выделенной передаче. По умолчанию это создаст простое правило для поиска передач с таким же названием, но вы можете использовать шаблоны и другие ключевые слова для автоматической записи передач, удовлетворяющих требуемому критерию.

Даже нам приходится отметить, что иногда в мире телевидения есть что посмотреть. [Разумеется, речь идет о зарубежном телевидении, – прим. пер.] Многие люди ценят *MythTV* именно за работу с телесигналом. Можно смотреть ТВ вживую либо записывать передачи для последующего просмотра, и функций для этого полным-полно. Например, при живом просмотре можно нажать P для приостановки передачи – Mythbuntu запишет программу с этого места; а вторично нажав P, вы сможете возобновить просмотр. Отстав от трансляции на пару-тройку минут, улучите момент и используйте курсорные клавиши для перехода вперед – это особенно удобно, если неохота смотреть рекламу.

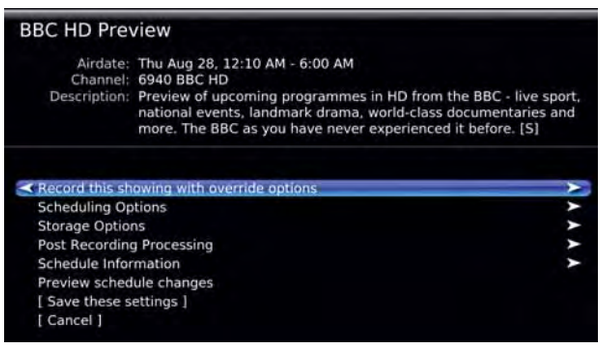
Нажатие R запишет всю передачу, причем с самого начала, если вы смотрели с начала: *MythTV* кэширует данные с момента старта просмотра канала. Нажмите M, чтобы вывести меню, откуда вы можете просмотреть программу передач, поменять масштаб и отношение сторон, а также подправить скорость воспроизведения, контраст и яркость картинки. Нажатие O выведет экранную информацию о текущей передаче, а курсорные клавиши переключают между каналами. Нажав пробел, вы создадите закладку, чтобы можно было перейти к этому месту, если вы вернетесь к записи. W сменит отношение сторон, а громкость регулируется клавишами квадратных скобок.

Живое телевидение – вещь полезная, но волшебная идея персональной видеозаписи освобождает вас от необходимости мчаться домой по расписанию: любую передачу можно записать и смотреть ее, когда захочется, а не тогда, когда угодно вещателю. Эта возможность спрятана в меню Управление записями [Manage Recordings], там вам нужно выбрать первую опцию, Планирование записей [Schedule Recordings].

> «Можно записать любую передачу и смотреть ее, когда захочется.»



› **Ставьте на паузу, проматывайте взад-вперед живые телетрансляции, пропуская рекламу и вновь и вновь наслаждаясь лучшими моментами.**



› **Порой *MythTV* может утопить простое действие в болоте опций. Однако поняв, что они делают, вы получите беспрецедентный контроль над вашими записями.**

Управление записями

В появляющемся меню охвачены шесть различных способов записи телепередачи. Первый – выбор ее в телепрограмме в представлении Live TV. Этот метод, вероятно, покажется наиболее знакомым: программы распределены по сетке, и можно начать запись, нажав на R на выбранном вами блоке. Если нажать на Enter, станут доступны другие опции записи, которые выполняют следующее:

» **Record only this showing** [Записать только этот показ] записывает выделенную передачу.

» **Record one showing of this title** [Записать один показ этой передачи] запишет эту передачу либо в настоящий момент, либо при ее повторе. Это полезно, если есть другие передачи, которые вы хотите записать.

» **Record in this time slot every week** [Записывать каждую неделю в это время] игнорирует название и информацию о передаче – и просто каждую неделю записывает то, что идет по заданному каналу в указанное время.

» **Record one showing of this title every week** [Записывать один показ этой передачи каждую неделю] пытается найти повторение той же передачи и записывает либо то, что вы выбрали, либо повтор.

Две следующих опции предусмотрены для ежедневных трансляций.

» **Record at any time on this channel** [Записать в любое время на этом

Интерфейсы и движки

MythTV пользуется моделью под названием «клиент–сервер». Это звучит пугающе, особенно если вы хотите всего лишь посмотреть телевизор, но зато дает большую гибкость. Серверная часть системы (движок, что отвечает за планирование и осуществление записи) совершенно отделена и отлична от интерфейса (части, используемой для воспроизведения). Другими словами, у вас есть несколько маломощных, малошумных устройств в разных комнатах, используемых для воспроизведения, и общая мощная машина-сервер, засунутая в погреб. Нередко интерфейсная и серверная части ставятся на одной машине, и мы сделали то же самое. Преимущество здесь в получении одного самодостаточного устройства, а если вам нужно больше интерфейсных клиентов для просмотра ТВ в других комнатах, вы можете просто добавить их к сети. Можно даже использовать окружение Mythbuntu Live для превращения любого компьютера вашей домашней сети в эпизодического клиента.

канале] запишет любые повторы этой передачи.

» **Record at any time on any channel** [Записать в любое время на любом канале] запишет передачу, независимо от того, транслируется ли она Dave или ITV 4.

Другие опции предоставляют альтернативные способы отслеживания передач, которые вы хотите записать. Поиск передачи [Programme Finder] – это алфавитный список всех передач в базе данных *MythTV*. Поиск по словам и спискам [Search Words and Lists] обеспечивает просмотр в базе данных по каналу, категории, названию, дате и времени, а Пользовательский критерий [Custom rule] позволит слепить какое-нибудь примитивное выражение, чтобы раздобыть ту самую передачу, что шла поздно ночью. Наконец, опция Ручное планирование [Manual Schedule] – это реализация в *MythTV* замысловатого метода программирования видеозаписи. Введите дату, время и канал, плюньте через плечо… авось ваша передача запишется.

Вернемся в меню: есть и другие варианты управления. Например, вы хотите регулярно записывать две передачи; вам пригодится опция Установить приоритеты [Set Priorities], чтобы, допустим, при одновременной трансляции сохранялся *Дом-2*, а не *Универ* [чур меня! – прим. ред.]. Просмотрев Ближайшие записи [Upcoming Recordings], вы узнаете, что собирается записывать Mythbuntu. Для удаления служит опция Удалить записи [Delete Recordings], но без нее легко обойтись: по мере переполнения диска *MythTV* сам их ликвидирует, начиная с наиболее давних. Если вы хотите, чтобы с любезной вашему сердцу передачей этого не произошло, отключите опцию auto-expire в окне Просмотра записей [Watch Recordings].

Просмотр записей

Завершив труды по записи передач на жесткий диск, можете плеснуть себе винца и что-нибудь просмотреть. Вам поможет пункт

› **Значки, расположение рядом с записью передачи, обозначают такие вещи, как высокая четкость и объемный звук, а также специальные опции *MythTV*, например, закладки.**

Просмотр записей [Watch Recordings] из меню Библиотека мультимедиа. Появляющийся экран показывает самые недавние записи, и динамический список наименований в левой части экрана можно быстро профильтровать по типу желаемой передачи. Например, если вы записывали *Дом-2*, выберите его слева, и увидите только этот сериал. Нажатием I можно поменять опции хранения и воспроизведения, а переключение в полноэкранный режим для выбранной записи делается нажатием Enter (функционально это почти идентично виду Live TV). Если вы прерываете просмотр на полпути, создается закладка, и при возврате к этой передаче не придется начинать все заново.

»

MythTV для бывалых

Кто бы мог подумать, что из телевизора можно столько всего выудить?

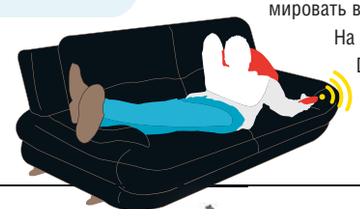
Создание DVD

Заархивируйте ваши записи на DVD — еще и с меню!

Раз уж Mythbuntu стал вашим репозиторием для мультимедиа любого формата, почему бы ему заодно не создавать архивные DVD? Это не только прекрасный способ высвободить ценное дисковое пространство вашей медиа-системы, но также способ делиться своими записями с другими и смотреть их на DVD-плейере. Путь к этому волшебству лежит через опцию Архивировать файлы [Archive Files], в меню Оптические диски [Optical Disks]. Выбрав Экспорт видео [Export Video], а затем Создать DVD [Create DVD], вы откроете нужную страницу.

Здесь указывается тип диска, который необходимо записать – он зависит от вашего оборудования, но чаще всего встречаются однослойные DVD. Следующая страница – для выбора файлов, предназначенных к записи. Пункт Добавить запись [Add Recording] откроет список телепередач, имеющихся на жестком диске в данный момент; Добавить видео [Add Video] отобразит видео; а Добавить файл [Add File] позволит вставить ваши собственные видеофайлы. На каждом экране обязательно поставьте галочку у файлов, предназначенных для записи. По мере добавления файлов, индикатор справа внизу экрана будет информировать вас о том, сколько места осталось на диске.

На следующей странице можно установить тему DVD-меню – она будет использоваться вашим DVD-проигрывателем при выборе нужной записи. Mythbuntu включает несколько тем, дублирующих его собственный интерфейс. Наконец, еще на одной странице можно указать параметры кодирования выбранных файлов, с тем, чтобы они уместились в отведенном месте или удовлетворяли заданным требованиям по качеству. Щелкните по кнопке Завершить [Finish] – и можете погулять пару часов, пока изготовляется диск.



› Выбор DVD: Mythbuntu умеет записывать одно- и двухслойные диски и создавать ISO-образы для копирования на другую машину.

Для ТВ

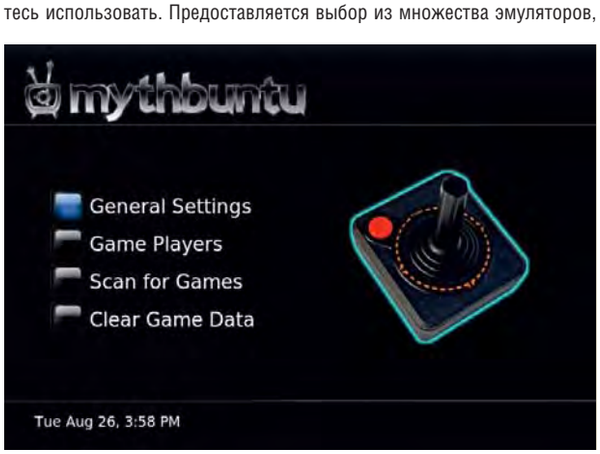
Быстрая запись
Файлы, записанные непосредственно с устройства цифрового телевидения, можно прожигать на DVD без дальнейшей обработки, сэкономив много времени и ресурсов CPU. Просто укажите в Профиле кодирования [Encode Profile] параметр Не перекодировать [Don't re-encode], и ваши выбранные записи запишутся на диск.

Игры

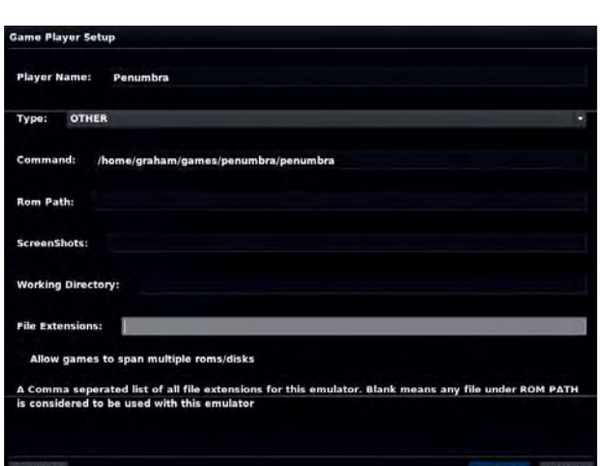
Не все ж только работать; без игр в Mythbuntu было бы скучно

Играть любят все, и Linux – прекрасная платформа для ретроигр, а не только для редких телепрограмм, которые достойны места на нашем компьютере. По умолчанию игры в Mythbuntu не включены, но есть удобный способ добавить свои собственные, чтобы превратить ваш скромный медиацентр в виртуальный эквивалент парка развлечений.

Страница настройки находится в меню Утилиты/Настройка > Настройка > Настройки мультимедиа > Настройки игр [Game Settings]. На странице Типовые игры [General Games] укажите, где будут игры храниться и как они будут отображаться, а на странице Игровые Проигрыватели [Game Players] отметьте движок, который вы собираетесь использовать. Предоставляется выбор из множества эмуляторов, и понадобится ввести команду его вызова. К счастью, Mythbuntu легко настроить как шлюз для всех Linux-игр, которые у вас найдутся. Просто выберите в качестве типа Другое [Other] и введите команду запуска игры. Затем можно запустить игру из Mythbuntu, а выйдя из нее, вернуться в интерфейс ТВ. Это идеально для небольшого расслабона, пока записывается *Планета вампиров*.



› Mythbuntu пригодится и для запуска ваших Linux-игр, и вам не нужно будет выходить из приложения и запускать его заново после окончания игры.



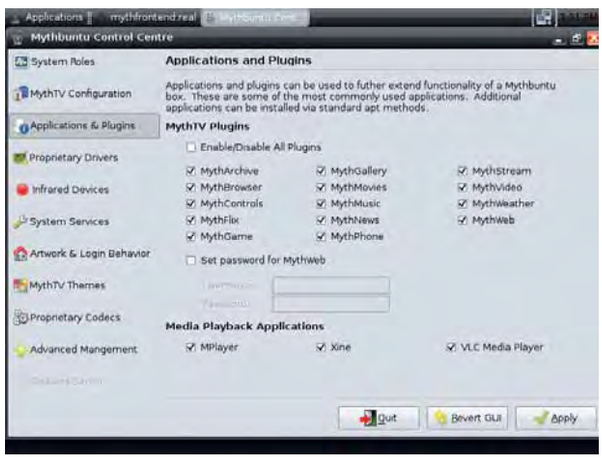
› Если вы перенастроите игру на использование USB-джойстика, даже клавиатура, подключенная к системе, будет не нужна. Просто выйдите из игры, и вернетесь в Mythbuntu.

Улучшенная настройка

Возиться с командной строкой не придется

Другой веской причиной использовать Mythbuntu вместо стандартной установки *MythTV* является включение настраиваемой конфигурационной панели под названием Центр управления Mythbuntu [Mythbuntu Control Centre]. Она запускается из Утилиты/Настройка > Настройка > Mythbuntu, и ее окно отображается поверх основного интерфейса пользователя.

Отсюда можно перенастроить графические драйверы, добавить инфракрасный пульт управления и установить проприетарные кодеки для воспроизведения DVD и Windows Media. Вы также можете включить системные службы – например, протокол удаленного рабочего стола VNC; активировать и отключать имеющиеся в Mythbuntu различные модули расширения *MythTV*, а также всевозможные темы. Страница System Roles позволит установить полное окружение рабочего стола Ubuntu или перенастроить медиацентр, сделав его интерфейсной машиной, подключаемой к серверу, расположенному где-нибудь в локальной сети.



› Это приложение переведет мороку с файлами конфигурации в разряд воспоминаний.

Дистанционная запись

Наша любимая функция

Web-интерфейс Mythbuntu, возможно, является лучшей его функцией. Благодаря ему почти все задачи Mythbuntu можно выполнять удаленно через web-браузер. Сюда входят просмотр программы телепередач, настройка записи, просмотр передач и видео, прослушивание музыки и рассматривание фотоколлекции. Страница с программой телепередач даже проще в использовании, чем та, что встроена в *MythTV*, так как вы можете легко пролистывать время, даты, щелкать по передачам и искать те, что вам интересны. Страницу с опциями записи также легче понять, чем ее эквивалент на телеэкране, а вот страницы настройки базы данных следует избегать: сделанные там изменения могут серьезно повлиять на стабильность вашей системы.

На страницу можно попасть с любого компьютера, находящегося в той же сети, что и медиацентр. Просто укажите вашему web-браузеру IP-адрес медиацентра, после которого напишите **mythweb** (например, **http://192.168.133.30/mythweb/**). IP-адрес медиацентра можно узнать, выйдя из интерфейса *MythTV* и щелкнув правой кнопкой по значку сети, расположенному справа вверху экрана. Оттуда выберите

› *MythWeb* позволяет производить практически любое действие, допустимое в *MythTV*, с любого компьютера, подключенного к Интернету.

Информация о соединении [Connection Information], и появится окно, где наряду с другими параметрами сети выведен IP-адрес. Если вы хотите получить доступ к *MythWeb* из Интернета, заставьте свой маршрутизатор перенаправлять запросы на порт 80 (web-сервер) на медиацентр, проследив, что этот порт не заблокирован ни маршрутизатором, ни вашим провайдером. Затем останется только указать вашему web-браузеру внешний IP-адрес вашего интернет-соединения. LXF

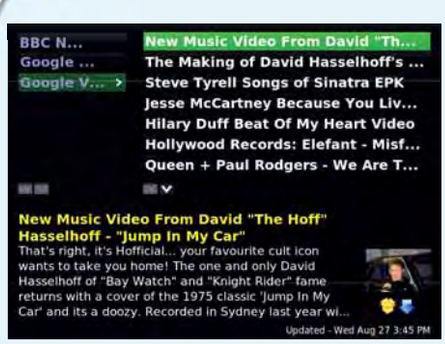
Для ТВ

PHPMyAdmin

Если вам когда-нибудь придется влезаать в *MySQL*-базу данных *MythTV*, пользуйтесь *PHPMyAdmin* вместо длинноющего интерфейса командной строки *MySQL*. Серьезно, вы еще скажете нам спасибо за этот совет.

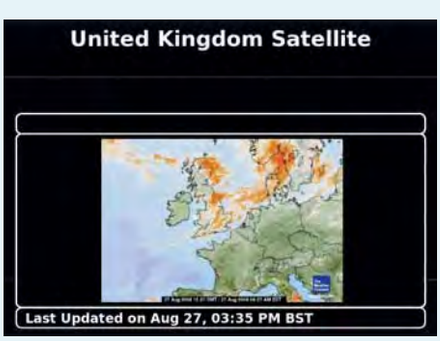
Информационный центр

Изучать все прелести MythTV можно неделями. В первую очередь взгляните на...



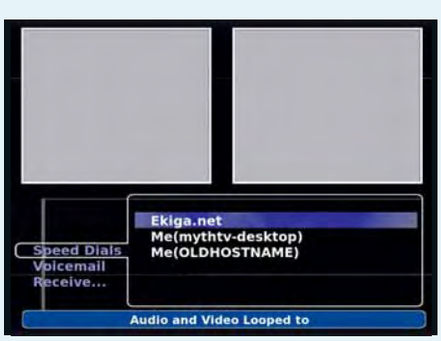
1 Ленты новостей

Mythbuntu может подписываться на RSS-ленты. В меню настроек Информационного центра [Information Centre] нужно сначала выбрать ленты, которые вы хотите просматривать. После этого при новом посещении страницы Ленты новостей [News Feeds] вы увидите список статей. Если статья содержит видео, нажатие Enter автоматически скачает его.



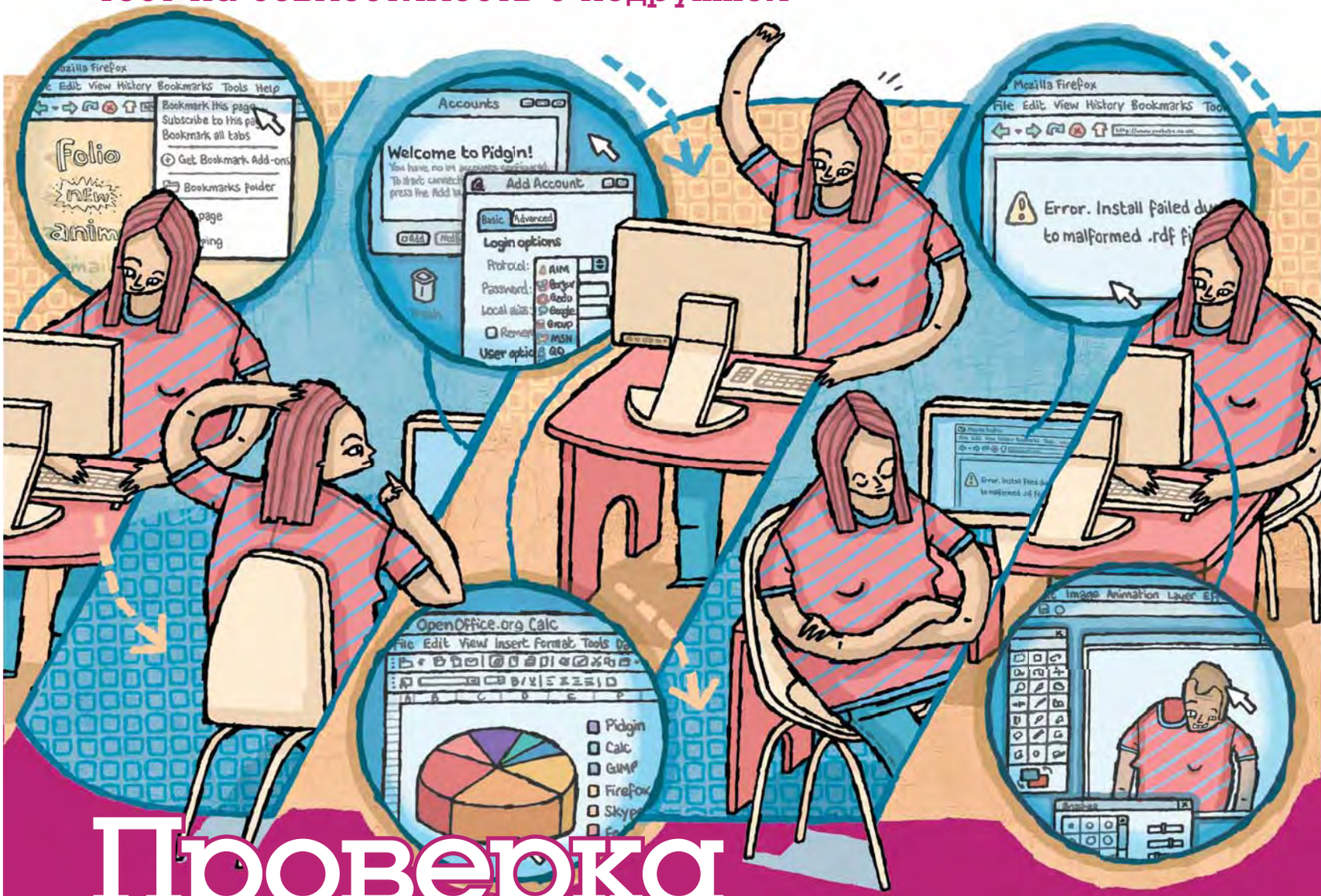
2 Погода

Сначала зайдите в Настройки экрана [Screen Setup] и активируйте одну из метеослужб, перечисленных слева. Выберите ваши единицы измерения и введите имя ближайшего к вам большого города. Выберите один из результатов – и готово. Теперь статус погоды отображает выбранный вами прогноз.



3 Видеоконференции

При наличии web-камеры, подключенной к вашему медиа-центру, вы можете создавать и записывать видеоконференции через интерфейс -. Он использует протокол SIP, т.е. вы сможете звонить всем, кто имеет стандартный SIP-адрес, а также по наземной линии, с помощью службы Free World Dialup.



Проверка на девчонках

Ученый-любитель **Алистер Рош** подвергает Linux самой придирчивой проверке: тесту на совместимость с его подружкой Эрин.

За последние 10 лет мир здорово изменился. Человечество наконец-то заполучило летающие автомобили, неисчерпаемый источник энергии и лекарство от рака. Ну, не совсем так, но Linux почти готов для среднестатистического ПК. А это ничуть не хуже. В некотором роде.

Прежде чем откупоривать бутылочку шампанского Canonical, не мешает кое-что прояснить. Linux все еще имеет репутацию слишком заумной, технической системы, предназначенной «только для хакеров». Этот миф нужно развеять как можно быстрее. Как? Испытав последние дистрибутивы крайне важным в научном мире тестом: на понятность подружке. Видите ли, старые проблемы несовместимости с оборудованием, бывшие когда-то корнем всех зол в Linux, исчезают – особенно теперь, когда крупные изготовители типа Asus и Dell принялись подлизываться к Туксу. На первый план вышли проблемы «обращения язычников» (они же – «пользователи Windows»), требующие создания как можно более дружелюбной, открытой и вдохновляющей атмосферы для работы. Этого нужно добиться до того, как Linux наберет критическую массу пользователей.

> «Эрин – среднестатистический пользователь ПК; таких мы и завлекаем в Open Source.»

Наш подопытный кролик – Эрин, девушка, представляющая собой типичного молодого пользователя ПК – намечен для «крещения» в веру Open Source. В обмен на участие автора в схожем эксперименте «Шопинг с бойфрендом», Эрин согласилась выполнить ряд заданий на свежеустановленной Fedora 9, в надежде выявить упущения по части удобства ее использования. Опыт работы на ПК у Эрин в основном ограничен применением компьютера для развлечения и выполнения заданий для университета: работа с почтой, *MS Office* и *Photoshop*, просмотр web-страниц и проигрывание музыки. Под Windows она справляется со всем этим без проблем. По части Linux, мы предоставили ей лишь имя пользователя и пароль для Fedora 9, и вот что у нас вышло.

Задача 1: Сделать закладку в Firefox

Как и следовало ожидать, у Эрин не возникло никаких проблем с выполнением этого задания. По умолчанию, *Firefox* находится на главной панели Gnome, что было для Эрин сюрпризом. Она понятия не имела об Open Source – просто знала *Firefox,* ибо это стандартный браузер в ее университете.

Кроссплатформенные приложения помогают среднестатистическому пользователю в освоении Linux; надеюсь, что по мере взросления открытого ПО подобных программ будет больше. Как бы то ни было, начало для Эрин оказалось хорошим.

Задача 2: Написать и распечатать письмо в OpenOffice.org

Найти *OpenOffice.org Writer* было легко: он находится в меню Приложения > Офис. При выполнении необходимых Эрин задач интерфейс *Writer* был весьма схож с таковым фактического стандарта, *Microsoft Word*: девушка запросто набрала и отформатировала письмо. Однако, как легко предугадать, проблемы возникли при попытке распечатать набранный текст.

После 10 секунд молчания приложение запустило Мастер устранения неполадок. Это было впечатляюще, ведь ничто не обескураживает более, чем компьютер, выдающий вам безликое сообщение об ошибке. Еще совсем недавно пользователю приходилось перелопачивать журналы в дебрях файловой системы, чтоб хотя бы понять, что произошло. К сожалению, Мастер не решил проблему, поскольку не обнаружил драйверов. Та же ситуация имеет место в Vista, но Microsoft хотя бы предоставляет ссылку на сайт Lexmark. Зная, насколько халатно некоторые изготовители относятся к поддержке Linux, лучше бы Мастер порекомендовал ключевые слова для поиска или форумы по тематике.

Первый рефлекс бывалого линуксоида – поискать ключ к разгадке в Сети, но Эрин и подобные ей пользователи о таком и не подозревают. О преимуществе наличия сообщества опытных пользователей Linux и его разработчиков следует кричать с крыш, чтобы неуверенные в себе новички не «открывали Америк» сами.

Задача 3: Записать CD-диск

Вы удивитесь: в эпоху скачивания всего подряд из Интернет наша Эрин все еще покупает музыку на компакт-дисках. Она легко конвертирует их в *iTunes* и заливает в свой iPod Nano. Когда Эрин попросили составить список задач, выполняемых ею на компьютере, в порядке их важности, эта оказалась в числе первых. Через секунду после того, как Эрин вставила CD в привод, Fedora спросила, что она хочет сделать, и предложила на выбор проигрывание в *Rhythmbox* или открытие в *Sound Juicer*. Последнее – просто, ясно и функционально: Эрин потребовалось только нажать на большую кнопку Извлечь, а *Sound Juicer* позаботился обо всем остальном.

› Быстрый и функциональный настолько, насколько это возможно, *Sound Juicer* воплощает философию Gnome.

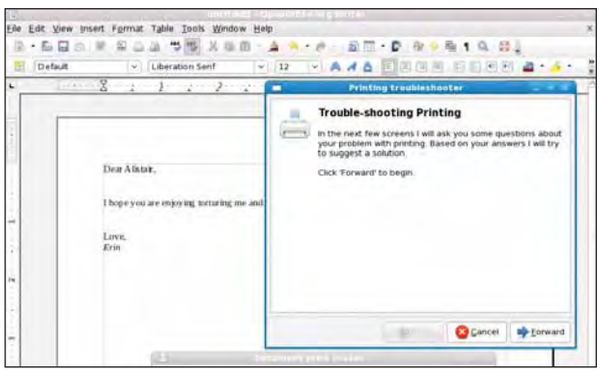
Треки сохраняются с использованием открытого кодека Ogg Vorbis. Эрин, как и большинство пользователей, знать не знает, ни что такое кодек, ни то, что музыка, проигрываемая на компьютере, может быть и не в формате MP3. С сохранением песен, таким образом, будет все хорошо, если Эрин ограничится их прослушиванием на данной конкретной системе; ну, а если она захочет воспроизводить файлы на компьютере своей подружки или на iPod? Они не зазвучат, а почему – будет непонятно.

Нет ничего плохого в том, что Fedora по умолчанию извлекает музыку с помощью Ogg Vorbis, но система должна информировать об этом пользователя при первом запуске *Sound Juicer* и предупредить о последствиях и о возможности выбрать вместо этого MP3. Если разработчики хотят добиться полной функциональности, им нечего прятать голову в песок, игнорируя доминирующие стандарты.

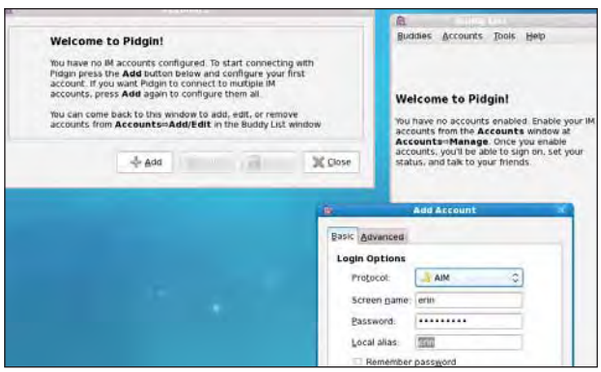
Задача 4: Отправить мгновенное сообщение

Для общения со друзьями Эрин использует клиент *Windows Live Messenger*. Ее первым импульсом было зайти на сайт Microsoft и скачать его. Поскольку это исполняемый файл для Windows, все попытки установить клиент не увенчались успехом, к удивлению и огорчению Эрин. Способ установки программ в Linux отличается от Windows и довольно прост, если его объяснить; но отличий достаточно, чтобы озадачить любого непосвященного.

Просмотрев список программ, Эрин нашла *Pidgin*, означенный там как «Клиент обмена мгновенными сообщениями». После запуска появился Мастер, предложивший ей добавить учетную запись. *Pidgin* – мульти- »



› Хоть это особо и не помогло, Эрин, по крайней мере, убедилась, что разработчики Fedora заботятся о решении ее проблем.



› Pidgin мог бы предоставлять функцию быстрой настройки для новых пользователей, как другие клиенты обмена быстрыми сообщениями.

Тест на совместимость с подружкой

протокольный клиент, и это запутало Эрин. Она не знает, что такое AIM (протокол по умолчанию), а тем более, что вообще такое «протокол». Кроме того, использование MSN не предполагает создания учетной записи – вы просто входите в сеть. Это на самом деле должно быть опциональным. Эрин вообразила, что надо создать учетную запись в сети 'AIM'! После трех неудачных попыток она нашла MSN и ввела свои данные. Кто смекнул, что первым делом она сказала своим друзьям нечто не совсем лестное о Linux, тот ох как прав!

Задача 5: Создать диаграмму в OpenOffice.org

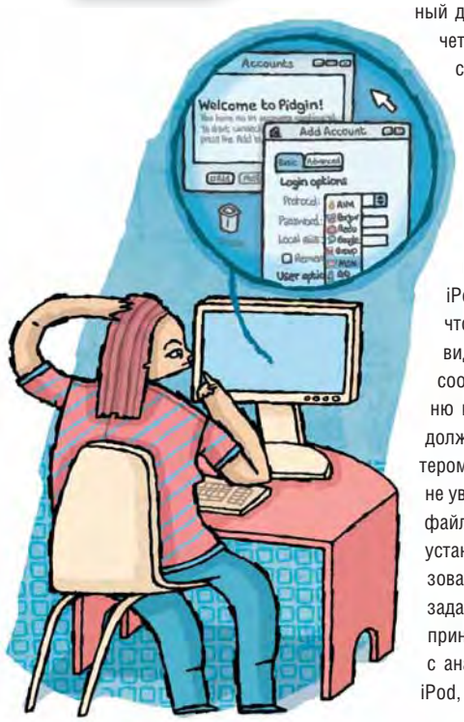
Задача «со звездочкой»

Эрин не приветствовала (да и не поняла) задание по перекомпиляции ядра. Все, чего мы добились, был удивленный и чуть обиженный взгляд. Но можно же мужчине помечтать, не так ли?

Без *OpenOffice.org*, Linux нельзя было бы назвать готовым решением, каким он ныне является. Пакет уже довольно хорошо развит, стабилен в работе и имеет все, что нужно для большинства пользователей. Попытка Эрин использовать *Calc* прошла так же гладко, как и работа с *Writer*. Поаплодируем Sun за тесное взаимодействие с сообществом Open Source. Хороший знак, что Эрин даже не удивилась отличной работе: на данном этапе она в ней не сомневалась.

Задача 6: Переписать музыку с CD на iPod

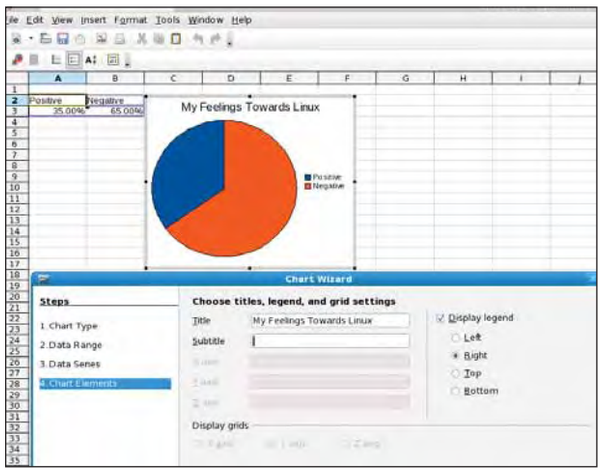
Эрин ежедневно слушает свой Nano и обновляет музыку на нем не реже раза в неделю. Она использует для этого *iTunes*, и затруднений никогда не испытывала. Когда Эрин подсоединила iPod, Fedora сразу распознала его и предложила открыть в *Rythmbox*. Спасибо Gnome, что он, подобно *iTunes*, свел музыкальный проигрыватель и менеджер портативных аудиоплейеров в одном приложении, иначе возникла бы путаница. Открылся *Rythmbox*, и ее альбом уже был в музыкальной библиотеке, хотя и конвертированный другим приложением – *Sound Juicer*. Прекрасно: четкая интеграция между приложениями рабочего стола в Fedora – это одно из преимуществ перед Windows.



Эрин выделила альбом и перенесла его на iPod, так же как в *iTunes*, но ее победная улыбка исчезла, когда она обнаружила, что выбранных песен на плейере нет. Она попробовала скопировать и вставить и перетащить свою музыку из папки в Nautilus, но это не помогло – iPod не поддерживает формат OGG. Поразительно, что *Rythmbox* даже не сообщает, почему файлы не видны после копирования – он вообще ничего не сообщает. В идеале, при попытке перенести песню или подкаст, записанный в Ogg Vorbis, на iPod, должен запускаться помощник (как в случае с принтером). Он сказал бы пользователю, что его попытки не увенчаются успехом, и научил, как конвертировать файл в распознаваемый формат, включая шаги по установке проприетарных кодеков. Конечный пользователь заинтересован только в выполнении своих задач при наименьших трудозатратах; то, что предприняла Эрин, предпринял бы любой пользователь с аналогичным опытом, и, оставшись без песен на iPod, склонился бы к отказу от Linux.

Задача 7: Приставить на фото ее голову к моему телу

Эрин, используя *Firefox*, нашла фотографии на Facebook и скачала их в свою домашнюю папку. Спасибо Fedora, что *GIMP* в меню приложений вразумительно значится как «GNU Image Manipulating Program», не то искать пришлось бы долго. Запустив *GIMP*, Эрин тут же осознала отличия интерфейса от *Photoshop*. После этого все легко: она открывает фотографию и вырезает свою голову, прилепляя ее к моему телу.



Если вы не бухгалтер, вы не заметите особой разницы между *MS Excel* и *OOo Calc*.

Потом она мне объяснила, что немало подивилась количеству качественных приложений в Fedora, и особенно сходству *OpenOffice.org* и *GIMP* с их коммерческими соперниками. Как бы то ни было, было бы неплохо предусмотреть интерфейс как в *Photoshop* для Windows или Mac OS X, на радость пришельцам с обеих систем.

Задача 8: Просмотреть видео на YouTube

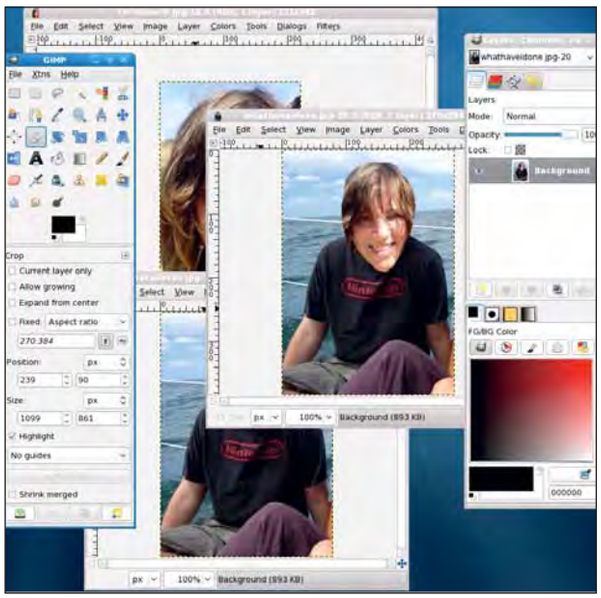
Эрин не из пещерного века – она знает, что такое YouTube, и навигация по сайту ей прекрасно удавалась. Но Flash в Fedora не установлен по умолчанию, поэтому *Firefox* предложил его скачать. Очередной помощник продемонстрировал ей лицензионное соглашение и загрузил нужные файлы – все легко.

Затем *Firefox* сообщил, что установка не завершена успешно из-за поврежденного RDF-файла. Вы, конечно, представляете себе гримасу на ее лице, до этого сиявшем оптимизмом. Она попробовала другой сайт, но безуспешно. Оборудование компьютера стандартное, и Эрин ничего не меняла. Базовая функциональность на самом деле должна много раз быть протестирована перед выпуском дистрибутива. Проприетарные форматы, пусть они неэффективны и принадлежат коммерческим организациям, весьма распространены. Почему бы в Fedora не объяснить пользователям при первой загрузке сайтов, что музыка и видео не будут работать так же, как на Windows, и как с этим бороться? Большинство перешедших на Linux или просто новичков, возможно, не умеют создавать скрипты, но уж читать-то простой и доступный текст они способны. Не так много надо и объяснять.

Ручная установка, предлагаемая *Firefox* после краха попытки автоматической, совершенно не подходит для пользователей типа Эрин.

На развилке дорог

Преимущество наличия разнообразных дистрибутивов в том, что каждый из них полезен в своей пользовательской нише. Всегда найдутся элитные хакеры, которым не нужны красивая оболочка, многословная документация или даже GUI. Если Fedora и подобные ей дистрибутивы жаждут распространения, пусть подумают о том, чего хочет от системы «человек с улицы». Так ведь ясно – чего: вращающийся 3D-куб для рабочего стола!



› Из-за отсутствия поддержки CMYK, *GIMP* не используется дизайнерами, но он вполне хорош для всех остальных.

Это просто ссылка на сайт Adobe, где дистрибутивные файлы предлагаются в виде архива **.tar.gz**, RPM и *Yum*, но нигде не сказано, что для Fedora нужны RPM. Туманные инструкции по установке на другой странице предлагают применить командную строку и не снабжены экранными снимками или сколько-нибудь подробными разъяснениями. Весьма разочаровывает, что обычная установка Flash не приспособлена к пользователям с опытом Эрин.

Задача 9: Позвонить по телефону через Skype

У Эрин, весьма охочей до общения, есть друзья по всему миру. Она звонит им задешево при помощи *Skype* и гарнитуры, как и сотни миллионов людей. У нее гарнитура от Sennheiser, которая вставляется в разъемы для наушников и микрофона на звуковой карте.

Эрин знает, что в Windows для установки приложения нужно зайти на web-страницу и скачать оттуда установочный файл. Возможно, некоторые читатели тут придут в ужас: неужто есть люди, способные не пойти в меню Система и Администрирование, чтобы найти пункт Добавить/Удалить приложение?! Но предположения Эрин типичны для новых пользователей.

К счастью, у *Skype* хватает дальновидности поддерживать большинство дистрибутивов Linux. А к несчастью для пользователей Fedora, поддержка добралась только до версии 7. Эрин нажала на ссылку, и действием *Firefox* по умолчанию было «Установить RPM». Все прошло в автоматическом режиме, и *Skype* обосновался в меню Приложения. Но не спешите радоваться: *Skype* отказался выполнить звонок из-за проблемы «с воспроизведением аудио». Эрин была в ступоре. Она убедилась, что звук выставлен на максимум, но кроме этого, не знала, что предпринять.

Понятно, что создатели Fedora не виноваты в том, что проприетарное программное обеспечение не всегда «дружит» с их дистрибутивом, но раз уж такие приложения используются массой людей, надо позаботиться о том, чтобы они работали. Наблюдаются значительные усилия разработчиков *Skype* по выпуску версий для Linux, и компания наверняка заинтересована в сотрудничестве с сообществом Fedora, чтобы *Skype* работал прямо из коробки.

Наши выводы

Одна из причин, почему люди тянутся к Linux – это тот факт, что разработчики системы стараются сделать взаимодействие с компьютером приятнее. Очевидно, не обходится без расхождений с работой в среде Windows. Трудно критиковать Linux, Gnome и Fedora за проблемы, возникающие у новых пользователей, которые привыкли к другой операционной системе, но ради практической цели нужны меры для более гладкого перехода.

Мы в сообществе Linux должны определить ожидания и нужды пользователя, и где Linux отстает в этом отношении. Многие проблемы вытекают из ориентации на высокую грамотность пользователя. Это очень распространено в Linux-сообществах, где и к разработке, и к оценке системы привлечены люди с техническим складом ума.

Безумием было бы делать вид, что Windows не существует и не является стандартом для большинства пользователей. Это не так, и дистрибутивы должны быть готовы к тому, что пользователи рассчитывают на работу определенным (и часто отличным от Linux) образом. Это вовсе не означает превращения Linux в клон Windows: просто следует задуматься об оповещении новичков: как, а иногда и почему, некоторые вещи работают в Linux иначе. Люди особенно противятся изменениям тогда, когда не понимают их смысла.

Смешно, конечно, советовать полному нулю в Linux написать **man man** в окне терминала и работать оттуда: этот подход не пробьет Linux дорогу к массам. Новые пользователи должны обучаться при помощи всплывающих окон и помощников. Почему бы не включить подсказки при загрузке или в процессе установки? Самая малость сведений дает неплохие результаты.

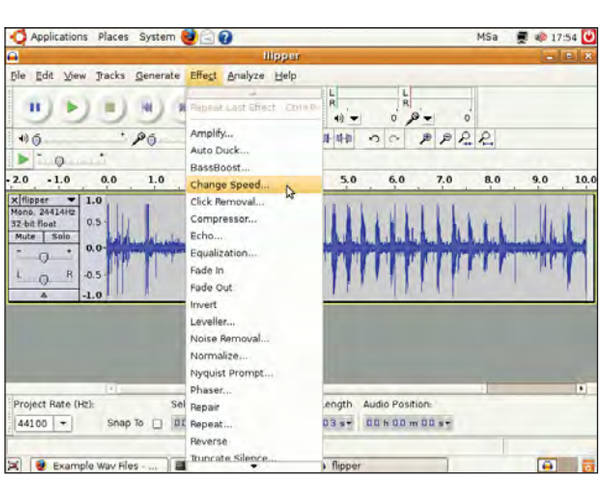
Открыт для всех

Экран приветствия, описывающий историю, терминологию и основные принципы, помог бы Эрин понять, что такое Linux, что он делает и как им пользоваться. Если гуру Linux не желают отвлекаться на подобного рода информацию, вовсе не сложно предусмотреть нечто вроде «режима новичка» только для неопытных пользователей. Ради популярности и массовости Linux стоит продумать дизайн приложений, дистрибутивов и вообще всего с точки зрения удобства конечного пользователя. Будем надеяться, что однажды кто-нибудь вроде Эрин сможет работать и играть под Linux без лишней мороки. LXF

Сборка — это просто!

Новая версия вашей любимой программы вышла без пакета для вашего дистрибутива. Что же делать? Да собрать ее из исходных текстов! **Майк Сондерс** расскажет все, что вам следует знать...

Сборка приложения из исходников – это старая школа, верно? Кому охота бороться с командной строкой, отлавливать зависимости и упрашивать компилятор *GCC* работать как полагается? Глупо заниматься этим в мире двоичных пакетов и сетевых репозиториев. В Интернете доступны тысячи пакетов, и все аккуратно собраны для наших дистрибутивов, аннулируя необходимость возиться с Make-файлами. Но это только кажется. При всем их удобстве, двоичные пакеты имеют множество ограничений, которые можно преодолеть, лишь собрав программу из исходных текстов. Вот некоторые преимущества ручной сборки:



Пакет *Audacity* из репозитория Ubuntu: да, неплохо, но можно добавить пару эффектов, собрав его из исходных текстов.

Злободневность Получайте программы сразу же после их публикации. КрутоеПриложение 2.1 уже вышло, а в вашем дистрибутиве все еще версия 2.0, и вам придется ждать следующего крупного релиза (если это только не самообновляющийся дистрибутив вроде Gentoo или Arch), чтобы обзавестись новинкой. Сборкой из исходных текстов вы сможете получить новейшую версию чего угодно сразу после выхода.

Функции В интересах стабильности издатели дистрибутивов при создании пакетов часто отключают экспериментальные функции, предоставляя вам более бедную (хотя и надежную) часть приложения. А идя путем исходных текстов, мы включаем дополнительные примочки – имеем право!

Оптимизация При сборке из исходных текстов вы получаете шанс сконфигурировать программу для вашей машины. Бинарные пакеты создаются для большого диапазона x86-процессоров. Используя определенные опции компилятора, вы можете создать специальное приложение именно для вашей модели процессора и извлечь дополнительную производительность из вашего ПК.

Доказать это проще, чем кажется; давайте рассмотрим, как оно работает. Навыки, приобретенные вами при чтении этой статьи, дадут хорошую основу для Linux (и Unix) карьеры, поскольку таким способом собираются почти все программы. Мы возьмем стандартный, немодифицированный вариант Ubuntu 8.04 и установим в нем звуковой редактор *Audacity*, включив при этом пару дополнительных опций. *Audacity* – хороший выбор, потому что в процессе его сборки могут возникать некоторые странные ошибки, а значит, мы научимся справляться с любой неожиданностью.

Подготовка

Прежде всего заполучим исходный текст *Audacity*. Вы можете найти его на сайте проекта в **http://audacity.sf.net**, или на **LXFDVD** в разделе **Звук**. Сохраните **audacity-src-1.2.6.tar.gz** на рабочем столе, затем откройте окно командной строки (Приложения > Вспомогательные программы > Терминал) и введите следующее:

```
cd Desktop
tar xfvz audacity-src-1.2.6.tar.gz
cd audacity-src-1.2.6
```

Первая команда переносит нас в каталог **Desktop**, а вторая извлекает сжатый архив **.tar.gz**. (Заметьте, что для архивов с суффиксом **.tar.bz2** вы должны использовать вместо нее tar xfvj). Наконец, мы перемещаемся в свежесозданный каталог с исходными текстами.

Введите ls, чтобы увидеть список распакованных файлов архива. Наряду с различными скриптами (подсвеченными зеленым) и подкаталогами (синего цвета), вы увидите файл **README.txt**. Всегда стоит глянуть в файлы с именами **README.txt** или **INSTALL.txt**, поскольку в них может быть полезная информация насчет сборки программы. Введите less README.txt, чтобы просмотреть файл, затем нажмите клавишу Q для выхода. В данном случае, нам не нужно читать все: можно перейти прямо к процессу компиляции.

Скрипт *configure* подготовит исходный текст *Audacity* к сборке. Введите команду

```
./configure --help
```

Она запустит скрипт настройки (./ требуется, потому что сценарий находится в текущем каталоге) и выведет все доступные опции сборки. Прокрутив терминал, вы увидите, что нам доступен огромный диапазон параметров. Но не дрожите – вполне можно принять значения по умолчанию. Хотя мы и собираемся сделать кое-какие мелкие изменения – во-первых, включим библиотеку *SoundTouch*: она предоставляет дополнительные инструменты манипуляции темпом и тоном, не вошедшие в стандартный пакет Ubuntu.

Далее, можно изменить место установки программы после сборки. Для этого используем со скриптом настройки опцию --prefix. Например, чтобы установить *Audacity* в каталог **/usr/local** (отдельно от других программ – так проще для поддержки), введите --prefix=/usr/local.

Как указывалось ранее, другим крупным преимуществом сборки программ из исходных текстов является выбор оптимизации. Установив переменные среды CFLAGS (для кода C) и CXXFLAGS (для C++) перед запуском скрипта *configure*, можно выбрать точные опции оптимизации компилятора *GCC*, которые мы хотим использовать. Это сложный предмет, но дополнительное увеличение скорости получить очень легко. Взгляните на эти команды:

```
export CFLAGS="-O3 -march=core2"
export CXXFLAGS="-O3 -march=core2"
```

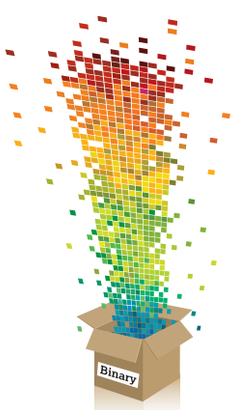
Так мы устанавливаем вышеупомянутые переменные среды с оптимизацией, соответствующей архитектуре Intel Core 2. Часть -O3 начинается с буквы O, а не с нуля, и устанавливает третий уровень оптимизации. (Большинство дистрибутивов собирают свои пакеты с -O2 для упрощения отладки и небольшого уменьшения размера исполняемого файла). Если у вас машина с чипом AMD64, то используйте -march=athlon64, а если вы не уверены, то используйте просто -march=i686 для более широкой совместимости. За полным списком поддерживаемых процессоров можете обратиться к man-страницам *GCC* (man gcc), но читать их нелегко!

> «Можете обратиться к man-страницам GCC, но читать их нелегко!»

Итак, мы настроили оптимизацию, и готовы к запуску скрипта *configure*. Вперед:

```
./configure --prefix=/usr/local --with-soundtouch=system
```

Тем самым мы велим скрипту произвести установку в каталог **/usr/local** и включить опцию *SoundTouch*, которая присутствовала в выводе ./configure --help, виденном нами ранее. (Часть 'system' опции *SoundTouch* говорит о том, что мы хотим использовать версию *SoundTouch*, доступную в дистрибутиве, поскольку поставляемая с *Audacity* содержит ошибки.)



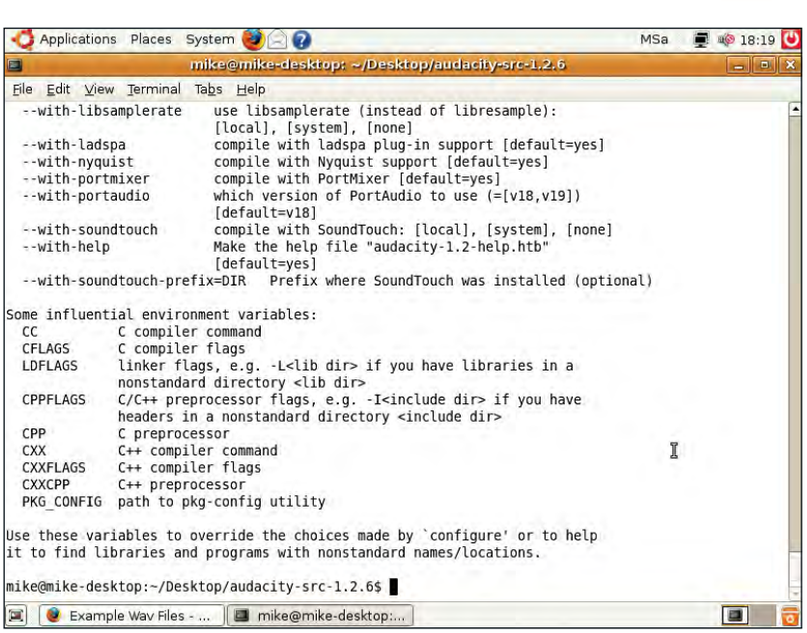
Зачем нужна сборка?

Если вы никогда до этого не видели исходного кода, вас может сбить с толку используемая здесь терминология, так что давайте разберемся с терминами и процессами.

Сборка (компиляция) – это процесс преобразования человеко-читаемого исходного кода (программы, записанной на одном из высокоуровневых языков программирования) в двоичные инструкции, используемые конкретным процессором данного компьютера. Для самых популярных языков программирования, C и C++, тяжелую работу преобразования исходного текста в двоичный исполняемый файл («бинарник») выполняет Коллекция Компиляторов *GNU* (*GCC – GNU Compiler Collection*): она читает исходные тексты и переводит содержащиеся в них инструкции C или C++ в машинный код вашего процессора.

Различные семейства процессоров имеют разные типы команд; например, при компиляции исходного текста приложения для работы на процессоре Intel, полученная программа не будет работать на процессоре PowerPC. Вам потребуется пересобрать ее для каждого типа процессора, на котором она будет использоваться.

Отметим, что для интерпретируемых языков, вроде Python, Perl или Ruby, каждая строка исходного текста выполняется на лету. Для этих языков нет никакого процесса компиляции – интерпретатор проходит программу строку за строкой.



Это результат работы ./configure --help – *GCC* показывает доступные нам опции настройки сборки.

Ловим зависимости

Ой, что-то не так. После запуска команды ./configure выскочила следующая ошибка: 'C compiler cannot create executables' [компилятор C не может создавать исполняемые файлы]. Для сборки двоичных файлов из исходных текстов необходимо установить инструментарий разработчика, так что введите:

```
sudo apt-get install build-essential
```

В Ubuntu это приведет к установке мета-пакета, включающего все инструменты, необходимые для сборки стандартных программ командной строки. Если у вас другой дистрибутив, поищите *GCC*, G++ и *Binutils* в своем менеджере пакетов и немедленно установите их, затем повторно выполните скрипт настройки. (Для новичков в командной строке подскажем экономящий время способ: нажмите клавишу «стрелка вверх», и ранее введенные команды вернутся).

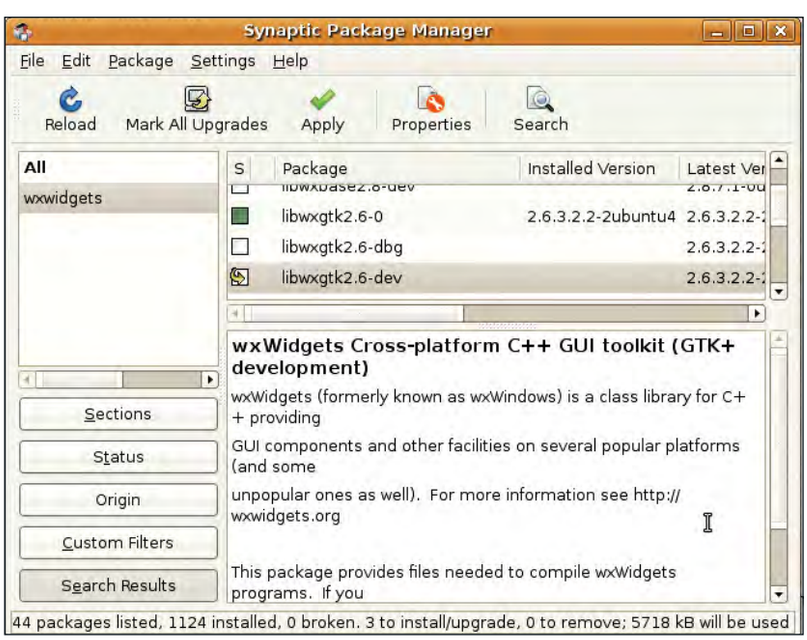
> «Многие программы Linux поставляются в двух версиях.»

Теперь скрипт проработает дольше, но споткнется и сообщит, что не смог найти в системе программу с именем *wx-config*. Для отрисовки интерфейса *Audacity* использует графический инструментарий под названием *wxWidgets*, а мы его еще не установили; потому и возникла ошибка. Итак, нужно извлечь *wxWidgets* из менеджера пакетов *Synaptic*.

Однако прежде чем браться за мышь, важно учесть, что многие программы Linux, особенно библиотеки и инструментарии, от которых зависят другие программы, поставляются в двух версиях. Одна из них – это библиотека времени выполнения, к которой могут подключаться программы, другая – набор библиотек разработчика и заголовочных файлов, используемых для сборки приложений. Итак, откройте *Synaptic* (в меню Система > Администрирование) и поищите *wxWidgets*.

Прокрутите появившийся список, и вы увидите *libwxgtk2.6-0*. Это библиотека времени выполнения, от которой зависят программы, использующие *wxWidgets* – она предоставляет реализацию вызовов *wxWidgets* средствами *GTK+*. Проверьте, установлена ли она, а также наличие *libwxgtk2.6-dev* (здесь содержатся необходимые файлы разработчика для сборки наших исходных текстов), затем примените изменения. (В других дистрибутивах эти пакеты будут иметь сходные имена, так что ищите *wxWidgets* или *wxgtk*. У пакетов разработчика имена почти всегда заканчиваются на *-dev* или *-devel*.)

Теперь перезапустите скрипт настройки. Вы увидите, что *Audacity* требует более старую версию *wxWidgets*, а именно 2.4.x! Это практически стандартная ситуация при сборке из исходных текстов, вот почему мы все это здесь и показываем; часто необходим метод проб и ошибок, чтобы точно выяснить, какие именно библиотеки и какие версии необходимы – и иногда это может утомлять. Итак, возвращаемся обратно в *Synaptic*, удаляем *wxWidgets* 2.6, и устанавливаем версию 2.4 (не забудьте пакет *-dev*).



› **Зависимости приложения можно установить парой щелчков в *Synaptic*.**

Магия Checkinstall

Если у вас несколько ПК или вы являетесь администратором большого парка корпоративных машин, то у вас, вероятно, нет времени для выполнения всех шагов сборки на каждом компьютере. К счастью, для этого есть инструмент, позволяющий создавать двоичные пакеты программ, собранных из исходных текстов. *Checkinstall*, который можно загрузить с http://tinyurl.com/2uodas (или взять в разделе Главное диска-приложения **LXFDVD**), создает пакеты Deb, RPM и .tgz для Slackware.

Установив *Checkinstall*, запустите скрипт **./configure** для приложения, которое вы хотите собрать, как было описано в тексте. Затем выполните **make** для выполнения сборки приложения, но не запускайте после этого обычное **make install**: вместо него, от имени root выполните **checkinstall**.

Тут вам будет задано несколько вопросов, например, про тип создаваемого пакета или хотите вы включить в него документацию или нет. По завершении работы *Checkinstall* вы получите двоичный пакет, который сможете установить на любом необходимом числе машин (если, конечно, на них тот же дистрибутив и установлены те же библиотеки).

Собираем код

Запустите скрипт *configure* еще раз, и вы увидите, что теперь все пройдет гладко. Для других программ может понадобиться обработать больше зависимостей, прежде чем удастся выполнить сборку – как уже говорилось, это всегда процесс проб и ошибок. Например, если *configure*-скрипт программы жалуется, что не может найти библиотеки разработки PNG, посмотрите пакет с именем вроде *libpng-dev*. Если вы собираете приложение, основанное на *Qt*, потребуется *libqt-dev*. Часто в файле **README** приводится полный список.

Команда, которую следует подать компилятору *GCC* – это (барабанная дробь)…

```
make
```

Вот так все просто. Утилита *make* обращается к только что созданному файлу Makefile, разъясняющему порядок сборки исходного кода программы. В зависимости от ее размера – и скорости вашей машины – процесс сборки может занять где-то от 10 секунд (для небольших приложений командной строки) до нескольких часов (для таких левиафанов, как *OpenOffice.org*).

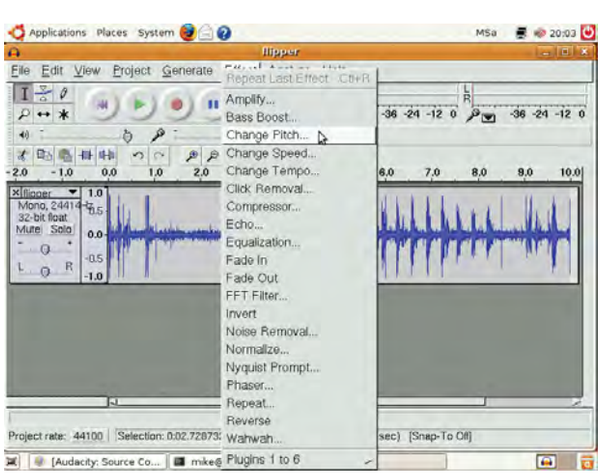
На этапе сборки вы заметите, что *GCC* вызывается с использованием настроек CFLAGS и CXXFLAGS, определенных нами ранее. Вы также можете увидеть предупреждения (warnings) о «нестрогом» коде, но о них беспокоиться нечего. Однако вы можете прийти в шок, когда процесс сборки вдруг остановится с ошибкой. Вы увидите, что **SoundTouchEffect.o** не может быть собран – означает ли это, что *SoundTouch* отсутствует?

Ну да, скрипт *configure* должен был бы предупредить нас об этом. Но не все в нашем мире совершенно – придется исправить проблему лично. Припомните, как мы в начале указали опцию --with-soundtouch=system для скрипта *configure*, вместо =local, потому что версия, поставляемая с *Audacity*, повреждена. (Вы и сами могли убедиться в том, что =local не работает, но мы малость сэкономили вам время!)

Итак, *configure* здесь нас немного подвел. Но дело за малым: откройте *Synaptic* и найдите и установите *libSoundTouch* (вместе с коллегой *-dev*). Скрипт настройки перезапускать не нужно – он завершился удачно, так что вновь введите **make** и следите за выводом. Теперь вы увидите, что не был найден FLAC. Ох! Запустите *Synaptic* и установите пакет библиотеки *libflac-dev*.

Введите ‘make’ еще раз, и вы увидите, что процесс сборки пошел дальше, но затем вновь остановился с целой серией ошибок, включая такую:

```
undefined reference to ‘soundtouch::SoundTouch::SoundTouch()’
```



› Ура! *Audacity* пересобран. Пусть интерфейс у него и на *GTK 1*, зато теперь у нас есть два классных новых эффекта.

Похоже, компилятор до сих пор не может найти *SoundTouch*, даже несмотря на то, что мы установили ее и на стадии *configure* предупреждений не было. Дожили! На последней стадии сборки, когда все файлы программы объединяются (компонуются или «линкуются») в исполняемый файл, *GCC* не понимает, где находится *SoundTouch*.

Необходимо изменить Make-файл, который управляет процессом сборки. Введите

```
gedit src/Makefile
```

Прокрутите до строки 31, где стоит ‘LIBS’ – это библиотеки, с которыми должен быть объединен (слинкован) *Audacity*. Добавьте в конец строки -lsoundtouch (регистр букв важен), сохраните файл и выйдите из редактора. Запустите *make* вновь. Вы увидите, что основной двоичный файл *Audacity* был собран корректно, но процесс уперся в новую стену: не нашлась утилита *msgfmt* для управления документацией.

На сей раз не обойтись без поиска в Google. Ищите ‘msgfmt’ и ‘Ubuntu’ (оба термина – в одном сеансе поиска, но не в кавычках, поскольку это не фраза); вы увидите, что другие пользователи наткнулись на те же проблемы при сборке программ. Пропажа, оказывается, сидит в пакете *gettext*, но – как это нудно! – у команды Ubuntu есть пакет *gettext* и пакет *gettext-base*. Последний был установлен как часть набора *build-essential*, тогда как важный для нас инструмент *msgfmt* содержится в первом. Может быть, когда-нибудь разработчики дистрибутива перестанут разбивать программы на триллионы двусмысленно названных пакетов; но хорошо уже и то, что решение быстро нашлось в Google.

Итак, установите *gettext*, чтобы получить *msgfmt*, и запустите **make** вновь. Как говорит Борат, большой успех! Вы вернетесь в командную строку без ошибок. Осталось только установить вновь собранный *Audacity* в файловой системе, так что введите:

> «Благодаря чудодейственной мощи сборки мы на шаг опередили Ubuntu.»

```
sudo make install
```

Это действие надо выполнить от имени пользователя root (администратора), потому и **sudo**. В не-Ubuntu дистрибутивах может потребоваться переключиться в режим root командой **su**, а затем ввести **make install**. И вы сможете запустить программу:

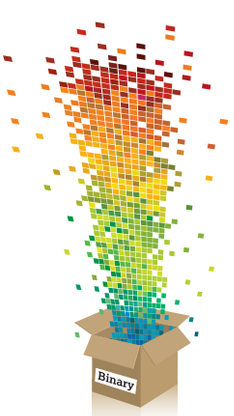
```
/usr/local/bin/audacity
```

Если надо удалить программу, выполните **sudo make uninstall**. Вы можете также удалить каталог с исходными текстами, когда захотите – откомпилированная версия теперь хранится в **/usr/local/**.

Шрифты – не фонтан

Вы можете отметить, что в этой версии *Audacity* шрифты выглядят не очень-то красиво. Это потому, что версия *wxWidgets* 2.4, включенная в Ubuntu, была собрана с *GTK1*, где нет сглаживания шрифтов. Пересборка *wxWidgets 2.4* с *GTK 2* – это еще одно упражнение, которое вы можете выполнить по завершении нашего урока! В любом случае, загрузите какой-нибудь звуковой файл и щелкните на меню Эффекты. Вы увидите две новые опции, которые не были доступны в базовой версии Ubuntu: Изменить тон и Изменить темп.

Благодаря чудодейственной мощи сборки мы на шаг опередили Ubuntu. Что, Шаттлворт, съели? Поскольку добавление поддержки *SoundTouch* ради двух наших дополнительных функций – дело хлопотное, неудивительно, что команда Ubuntu на него плюнула. Сборка большинства программ, за которые вы возьметесь, пройдет гладко или потребует лишь несколько очевидных зависимостей, но мы навидались всяческих вариантов возможных проблем, так что теперь вы обучены и готовы к любым неожиданностям. Теперь пора за сборку! LXF



30 дней с Haiku

Весна хранит Haiku
Расцвел синий ирис
Типа поэзия

(Грэм Моррисон, 2008)

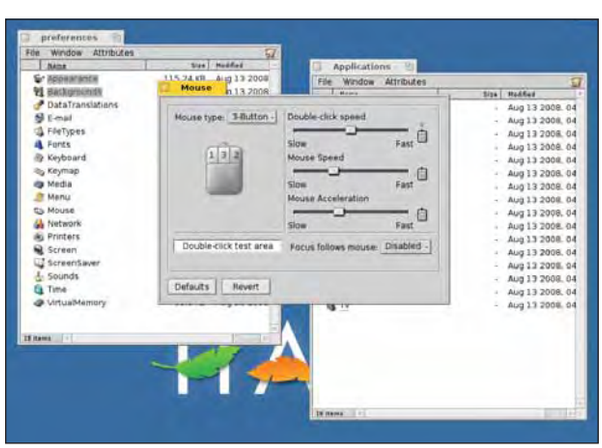
Haiku – свободная операционная система и альтернатива Linux. Она отметила свой седьмой день рождения 18 августа и все еще активно разрабатывается. Haiku никогда не считалась законченным проектом, но на сегодня она достаточно стабильна для повседневной работы. А главное, очень интересна. Дизайн Haiku очень похож на BeOS – но я никогда не работал в BeOS. Я не знаю, есть ли в ней браузер, файловый менеджер или даже командная строка. Я понятия не имею, как устанавливать пакеты и есть ли они. Вот моя история.

Помню, что BeOS мне очень понравилась десять лет назад, но и компания, и ОС скончались до того, как я успел ее попробовать. К счастью, BeOS произвела впечатление на тех, кто успел; как в случае с почтенным Commodore Amiga, предпринято несколько попыток воскресить ее. И здесь в дело вступают Haiku и открытое ПО. Выпущенная по лицензии MIT, Haiku – достойный преемник BeOS. Она не только совместима с исполняемыми файлами последнего релиза BeOS, но и содержит значительные улучшения по сравнению со своей предшественницей.

Поэзия на марше

Почему, если Linux – настолько мощная и полная возможностей операционная система, кто-то может предпочесть ему, возможно, худшую и менее популярную альтернативу? Ответ: а почему нет? Большинство из нас выбрали Linux тогда, когда все то же самое было справедливо и в отношении нашей любимой ОС. Не так уж давно Linux работал нестабильно и имел серьезные ограничения, и с приложениями, которые мы сейчас принимаем как данность, хватало проблем. Многие из тех, кто тогда однажды попробовал Linux, зареклись это делать и вернулись в Windows. Но без этого опыта они могли бы никогда не прийти в Linux снова. То же справедливо и для всех альтернативных ОС, включая Vista и OS X, поэтому я и обращаю свое внимание на Haiku. Я проведу 30 дней с этой открытой альтернативой Linux. Те, кто читал наше Сравнение открытых ОС в LXF105 (в нем фигурировали шесть вариантов), могут полюбопытствовать, почему я выбрал Haiku, а не победителя – Syllable. Ответ таков: для Haiku разработано больше приложений, следовательно, шире поле для деятельности.

Сперва, однако, признаюсь, что не собираюсь работать только в Haiku. Иначе я не смогу заниматься своими обычными делами. Вместо этого я запущу Haiku на виртуальной машине в Linux и буду использовать ее для всего, что смогу. Надеюсь. На следующих четырех страницах я опишу свое путешествие.



› День 1: Наш первый порт прибытия – окно «Настройки мыши». Здесь мы уменьшаем скорость перемещения курсора и возвращаемся обратно на рабочий стол.

Первые впечатления

После загрузки Haiku все выглядит жутко, как году в девяносто втором. Есть масса основных цветов, контур активного окна обведен желтой линией, а фон с пером логотипа – ярко-синие. Я обвыкся с темой BeOS из KDE, но это ничуть не помогает мне чувствовать себя как дома. Это освежает давние воспоминания о настройке шрифтов, установке иконок приложений и оставленной на ночь загрузке 100 КБ в древней сети JANET. К счастью, сеть ушла далеко вперед, а внешний вид рабочего стола теперь легко исправить.

Моя первая настройка интерфейса – изменение параметров мыши: она слишком быстрая, а при запуске системы в виртуальной машине еще и ускоряется. Панель ее настроек, упрятанную в правом верхнем углу экрана в странном доке «перо», я нашел легко, но сговориться с мышью оказалось далеко не просто.

Я понемногу передвигал регуляторы и возвращался из меню настроек мыши на рабочий стол, и тут система упала в первый раз. Хотя курсор мыши еще шевелился, рабочий стол не отвечал. Серьезное падение системы в первые пять минут работы напоминает мне времена KDE 4.0. Но, в отличие от KDE, перезагрузка Haiku – дело нескольких секунд.

Продравшись кое-как через эти трудности, я решил побродить по Сети и вернулся обратно к голубому перу в правом верхнем углу рабочего стола – единственному аналогу меню, который видел. Щелчок по нему радостно открывает набор других меню в стиле *Xfce*, одно из которых называется Applications [Приложения]. Здесь-то и сидел бы web-браузер – если бы был установлен. Вместо него в этом меню – список из дюжины технических демо-пунктов, ни один из которых ничем не обогащает рабочий стол.

С горя я открыл терминал и набрал ‘links’ и потом ‘lynx’, но в подсказке *Bash* не было и проблеска понимания. Я окончательно завис, и все, что смог придумать, было переключиться в Linux и использовать тот браузер, который точно работает. Так я и сделал.

День 7: Инструкции по установке и запуску *Firefox* удалось найти только благодаря форуму.

Google Documents работают отлично, и я впервые могу загрузить что-то в графической программе. Не поддерживаются Java и Flash, но это, наверное, и к лучшему: сэкономлю время от просмотра клипов начала восьмидесятых на YouTube. Web-серфинг на рабочем столе Haiku работал практически без проблем, и я продолжал в том же духе несколько дней, до тех пор, пока не захотел поглубже погрузиться в ОС.

> «Я завис, и все, что я мог придумать — это вернуться в Linux.»

Скудная документация

Сначала я попробовал найти онлайн-документацию по Haiku, но разработка Haiku покамест в стадии «свободного падения», а это означает, что документации для обычных пользователей нет – только для технарей. Осталось открыть Google и искать там ‘haiku getting started browser’. Это сработало, и первой же ссылкой была ссылка на форум по Haiku, где новый пользователь задавал тот же самый вопрос.

Очевидно, можно воспользоваться браузером *NetPositive* с первого установочного CD BeOS (которого у меня, ясное дело, нет) или загрузить подходящую версию *Opera* или *Firefox*. Я выбрал последний вариант, но он оказался непростым: нужно загрузить скомпилированный двоичный файл *Firefox* и создать несколько переменных окружения.

Раз напрямую загрузить страницу в Haiku нельзя, приходится скрупулезно копировать команду символ за символом в терминал Haiku. Но через несколько минут я уже влез в каталог *Firefox* и запустил исполняемый файл. Он не очень удобен, но приезжая в новую страну, подобных проблем и ожидаешь.

Оказывается, *Firefox* активно разрабатывается для BeOS; этот проект носит название *Bezilla*, потому что текущая версия не дотягивает до стандарта брэнда *Firefox* от Mozilla. Отсюда следует, что *Firefox* для BeOS всегда будет отставать от версии для Linux. То же самое справедливо и для Java, которая на данный момент не портирована в BeOS (хотя небольшая команда разработчиков недавно взялась за эту задачу).

После запуска web-браузера жизнь становится светлей. Теперь я могу хоть как-то использовать Haiku. *Firefox* немного тормозит, но в остальном это тот браузер, который мы знаем и любим. Google Mail и

»

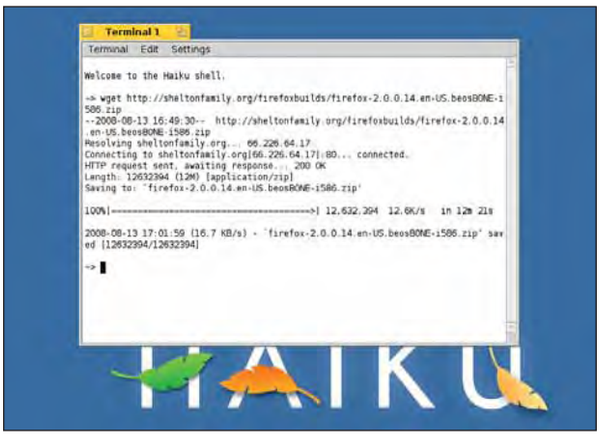
BeOS

BeOS, операционная система, на которой основана Haiku, разработана компанией, основанной бывшим исполнительным директором Apple Жаном-Луи Гассе [Jean-Louis Gassée]. В его время Apple сопутствовали успехи и неудачи, отражающие переменчивый процесс развития компании в восьмидесятых. И, как Стив Джобс [Steve Jobs], который сам покинул Apple в 1985 году, чтобы создать Next Computer, Жан-Луи Гассе ушел из Apple в 1990 году с высокими идеалами новой ОС, которую он называл просто ‘BeOS’. Компания Be Inc., занимавшаяся разработкой новой операционной системы, создавала ПО, которое должно было работать на особой аппаратной платформе – злополучном BeBox. Когда BeBox потерпел неудачу, BeOS адаптировали для запуска на архитектуре PowerPC, которая в те времена преимущественно использовалась на компьютерах Apple. В какой-то момент казалось даже, что Apple купит BeOS как замену собственной ОС, но Стив Джобс опередил Гассе, продав Next Apple и переформировав компанию в процессе. Однако BeOS получила достаточную поддержку сама по себе, поэтому мы сегодня и говорим о ней. И благодаря Haiku вы можете попробовать ее сами и сделать свои выводы.

Бенуа Шиллингс [Benoit Schillings] был вторым инженером, нанятым Be.

Игры

Потратив несколько дней на серфинг, я решил, что пора заняться чем-нибудь поинтереснее. И это означало установку дополнительных программ. К счастью, есть портал, на котором можно найти все, связанное с BeOS: его адрес **www.BeBits.com**. Это один из тех закоулков, о существовании которых в Интернете вы не подозреваете, пока нет причины их посетить. Пользователи Windows, должно быть, думали то же самое о Linux, пока не поискали его в Google и не осознали, что Интернет и есть Linux. BeBits – временна́я капсула для приложений, утилит и игр BeOS, многие из которых, я надеюсь, запустятся в Haiku без особых проблем. Так или иначе, это мысль.



› День 16: *Wget*, кажется, единственная уступка сетевым средам, но спасибо и за то, что сетевой стек является частью ОС.

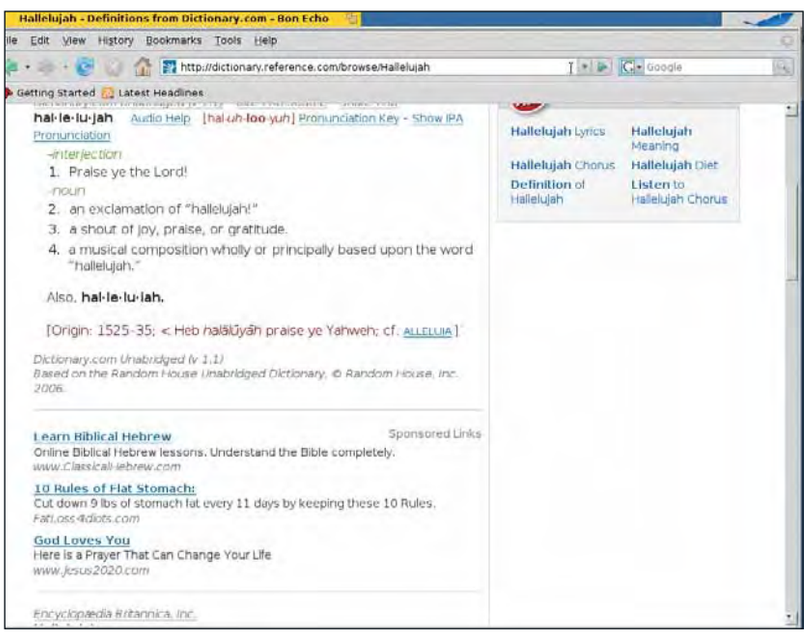
Серьезное дело

Ретро-стиль рабочего стола вызвал у меня желание поиграть во что-нибудь классическое, и я поискал на сайте аналог *Tanks* для BeOS, думая, что простота – хорошее начало. Нашлось несколько игр (включая классическую *Barrage*), но я выбрал примитивную классику *TankGame*. Увы, для ее загрузки нужно было зарегистрироваться на сайте.

Так как выбор у меня был большой, я просто вернулся на BeBits и попробовал другую игру – пресловутый *Barrage*, из-за хорошего рейтинга и размера всего в 709 КБ (или это означало «одну дискету»?). Но ссылка не работала. Я нашел на этой странице еще несколько дохлых ссылок и забытых программ, и осталось только поискать что-нибудь наугад.

Пройдясь по списку игр, я выбрал *Alien Attack 2*: это название я помнил со времен Acorn Electron, что и объясняет ее размер – всего 40 КБ и возраст – примерно 10 лет. Через доли секунды zip-архив был уже на рабочем столе, и, щелкнув на нем правой кнопкой мыши, я его распаковал. В Haiku приятно то, что часто приложения можно запускать напрямую. На рабочем столе даже появилась новая иконка, и по клику на ней игра стартовала. В маленьком окне. И не в меру быстро. Но игра была довольно милая, и мне осталось попробовать что-нибудь еще.

> «Большинство пользователей Haiku, похоже, считают ее нормальным рабочим столом.»



› День 10: Удивительно, какую роль завоевал в нашей жизни скромный web-браузер, с помощью которого ныне решается множество задач.

На сей раз я зашел в раздел со статистикой BeBits и взглянул, что загружают остальные пользователи. Самый большой рейтинг был у двух кроссплатформенных эмуляторов, *Qemu* и *ScummVM*. Оба прекрасно делают свою работу, но едва ли являются BeOS-эндемиками. Далее в этом списке идут *Vim*, музыкальный проигрыватель и BeOS-версия конвертора DVD *Handbrake*. Похоже, большинство людей используют Haiku как полноценную замену Linux, делая в нем то же самое, что и мы в своей ОС.

Есть и огромное количество пакетов с драйверами, включая утилиты для видеокарт Nvidia и ATI, а также простой драйвер Ethernet. Я порадовался, что работаю на виртуальной машине и не должен беспокоиться о совместимости оборудования. Здесь также можно найти почти все игры и приложения с открытым исходным кодом, которые есть в среднем менеджере пакетов Linux – выходит, что, несомненно, есть множество людей, которые работают с BeOS-версиями популярных программ.

Теплее...

Тут я наткнулся на сайт Haiku Ports (**http://ports.haiku-files.org**), который, похоже, представляет собой главный репозиторий современных проектов BeOS, которые переносятся с других платформ и поддерживаются на различных производных от BeOS системах. Таковые перечислены на главной странице проекта и включают как оригинальную BeOS, так и Haiku. Но интересным открытием оказались другие похожие альтернативы, в число которых входит Zeta.

Это коммерческое продолжение разработки BeOS, возглавляемое компанией YellowTab GmBH. Если я что и узнал о BeOS, так это что данная ОС вызвала массу инициатив, поддерживающих ее жизнь. Zeta привела меня на сайт **www.zeta-games.com**, с гораздо более современным подходом к BeBits. Оттуда я загрузил SDL-библиотеки. Распаковка этого архива воссоздала каталог **/home** на моем рабочем столе, и я вообразил, что для успеха установки нужно было перезаписать настоящий **/home**. А на самом деле нужно было скопировать so-файлы в каталог **/boot/home/config/lib**. Потом я попробовал несколько игр; они были полностью идентичны своим Linux-аналогам. В Haiku многое будет вам знакомо.

Офисная жизнь

После целой недели баловства с системой мне захотелось чего-то более общественно-полезного. К сожалению – и это самая большая проблема Haiku – соответствующего ПО очень мало. Есть древняя версия *AbiWord* для BeOS, которая даже официально поддерживалась какое-то время, пока ОС не скончалась. Но она слишком стара, чтобы воспринимать ее всерьез.

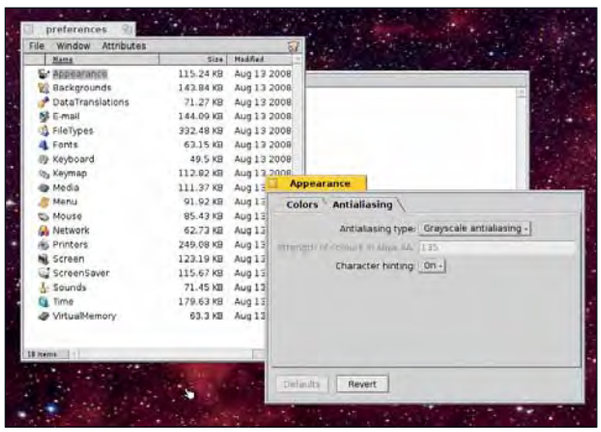
Есть несколько проектов, пытающихся портировать самую свежую версию *AbiWord*, включая 'pre-alpha' релиз *AbiWord 2*, но я не смог заставить ее работать, и, похоже, за последний год проекты совсем не развивались. Вы также можете урвать старый набор программ *GoBe Productive*. Одно время даже ходили слухи, что компания, которая разработала это ПО, хочет купить операционную систему, но похоже, что в те дни она зарабатывала больше на продаже приложений для Windows. Как я уже говорил, с Google Documents можно работать в *Firefox*, но это не лучший вариант для тех, кто привык к «родным» приложениям Linux.

Я не могу критиковать Haiku. Она находится в стадии развития, и никто не объявлял, что промышленный уровень стабильности уже достигнут. Но Haiku близка к этому, и невольно ощущается, что наличие документации для пользователей и система пакетов сильно ее продвинули бы. Последние версии *AbiWord* и *Firefox* для Haiku – все, что нужно для небольшого офиса, и Haiku можно будет установить практически на любой свободный компьютер в организации. Она также шустрее большинства ОС, включая Linux, а ограничения рабочего стола не столь важны.

С KDE на Xfce

Если вы решите пожить в мире Haiku, ее рабочий стол окажет сильнейшее влияние на вашу повседневную жизнь. Это как переключиться с KDE на Xfce, например. Важный компонент любого рабочего стола – файловый менеджер, и BeOS очень напоминает мне времена Amiga. Подобно пространственному режиму *Workbench* и Gnome, каждая папка открывается в новом окне, как и приложения и настройки. Границы окон достаточно узки, и пространство рабочего стола используется очень эффективно.

Это было важно десять лет назад, когда жилплощадь на экране дорого стоила, но в эпоху дешевых TFT-мониторов с высоким разрешением выглядит скаредно. Например, крошечный док в правом верхнем



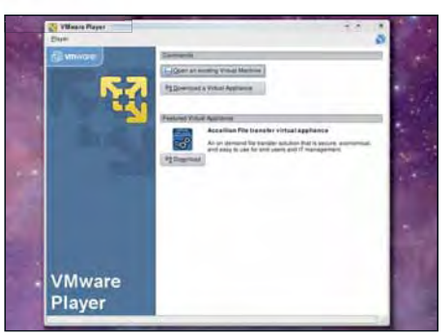
› День 22: В Haiku есть масса новых возможностей, включая аппаратное 3D-ускорение и сглаживание текста.

углу окна похож на старое приложение *Amiga Tool Dock*. И ко всей этой эффективности придется привыкать, а заодно и к ряду других мелочей. Как и в Apple OS X, приложения не завершаются по щелчку на иконке «закрыть» в левом верхнем углу окна, а сворачиваются в небольшую панель, и их можно снова открыть в любой момент. Чтобы завершить приложение, нужно кликнуть на иконке приложения на этой панели и выбрать пункт Close [Закрыть] всплывающего меню.

Компенсацией за этот минимализм является скорость. Окна быстро открываются и перемещаются. Большинство приложений и утилит загружаются моментально (исключение – *Firefox*), и систему можно полностью перезагрузить за 10 секунд, на достойном оборудовании – даже быстрее. Это рабочий стол, который нам всегда хотелось иметь.

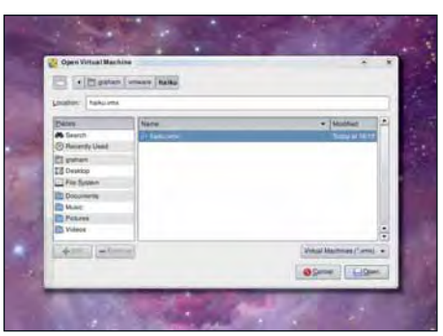
После нескольких недель жизни Haiku на моем настольном компьютере погружение в нее стало казаться мне на удивление освежающим. Она показывает, что не все на свете одинаково. Мир свободного ПО не ограничивается ядром Linux, и альтернативные ОС типа Haiku предоставляют нам неплохой выбор, напоминая, что Linux – еще не начало и не конец мира свободного ПО. А ведь это не так уж плохо?.. LXF

Шаг за шагом: Двинем в Haiku!



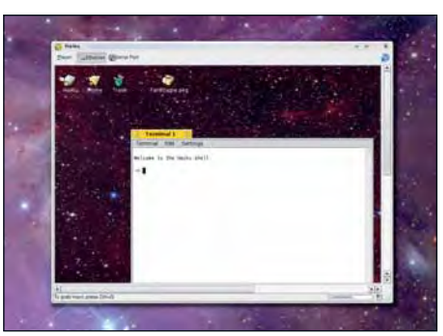
1 VMware Player

Установите и запустите бесплатную программу *VMware Player*. В дистрибутивах типа Ubuntu ее легко найти в менеджере пакетов, что позволяет добавить ее за пару щелчков мыши.



2 Найдите Haiku

Скопируйте и распакуйте образ виртуальной машины с Haiku на локальный диск, кликните Open Existing Virtual Machine [Открыть существующую виртуальную машину] в *VMware* и укажите каталог Haiku.



3 Запустите виртуальную машину

VMware спросит у вас о виртуальном последовательном порту, используемом Haiku для отладки. Через несколько секунд Haiku стартует, и перед вами появится рабочий стол.

Что за штука... DjVu?

Марко Фиоретти представляет формат, в который когда-нибудь обратятся все старинные свитки и современные киноафиши.

Вроде я такое название слышал… дежа-вю… Не напомните?

DjVu – это специальный формат файлов для сжатого хранения цифровых документов, изначально разработанный в AT&T.

Ну, обрадовали – мало нам JPEG, GIF, PNG и прочих? Зачем еще и DjVu?

Этот формат специально предназначен для сканирования фотографий и бумажных документов, для целей архивирования и онлайн-публикации.

Короче, обычному пользователю это без надобности?

Если ваши потребности ограничиваются альбомом цифровых фотографий – пожалуй, да. А вот ученые и многие другие люди – от любителей комиксов до коллекционеров живописи – явно имеют причины заинтересоваться этим форматом.

> «Хотите купить DVD с ПОЛНОЙ подшивкой любимого журнала?»

Это почему же?

Принято думать, что в сегодняшнем мире все цифровое, но на самом деле почти вся документация хранится либо на бумаге, либо на другом аналоговом носителе.

Какие документы вы имеете в виду?

Вспомните о старинных книгах, газетах и картах – уйма документов, от египетских свитков до первых изданий Бэтмена 40-х годов.

А что, DjVu-архивы уже где-то существуют? Приведите парочку примеров…

Их масса… не знаю даже, с чего начать… например, можно приобрести DVD с высококачественными репродукциями архивов журналов *Spectator* или *Rolling Stone*, но, вероятно, куда больше вас заинтересует бесплатное издание Иллюстрированного Шекспира на www.leoyan.com/djvu-editions.com/SHAKESPEARE/COMPLETE/toc.php. Демо, показывающее, на что способен DjVu, есть на http://djvu.org/gallery.

А малые предприятия и индивидуальные пользователи? Им DjVu пригодится?

DjVu мог бы стать недорогим способом онлайн-представления товара, чтобы покупатель мог как следует рассмотреть его перед покупкой.

Что касается индивидуальных пользователей, опять-таки подумайте об ученых: они могли бы разместить на своих сайтах сотни рукописных математических формул. Если это, по-вашему, нетипичный вариант, вспомните о тех, кто публикует свои рисунки или акварели на Flickr и вынуждены мириться с ограничениями JPEG. Поверьте, рынок найдется!

Допустим… но разве не для этого создан PDF?

PDF хорош, если нужно распознать и сохранить бумажную копию файла, но он лишен многих достоинств DjVu. Многие вещи, о которых я упоминал выше, не существуют в цифровом виде просто потому, что их практически неподъемно перевести в любой из существующих форматов, включая PDF.

А в чем тут дело?

Все упирается в системные ресурсы: читаемые JPEG, GIF или PNG-сканы могут быть настолько «тяжелыми», что их архивирование на сервере, не говоря уже о скачивании, при нынешнем состоянии коммуникаций просто неосуществимо.

Тогда как DjVu…

…может быть до 10 раз меньше аналогичного JPEG, и до восьми раз – PDF. Подробное слайд-шоу по этой теме можно увидеть на http://djvu.org/resources/djvuvsother. Шоу, естественно, в формате DjVu, поэтому дочитайте эту статью до конца, иначе вам его даже не открыть. А если вы просто хотите сравнить DjVu с JPEG, прочтите статью на http://djvu.org/resources/jpgvsdjvu или зайдите в галерею 'DjVuPhoto vs JPEG' на www.djvuzone.org/djvu/photos/jpgvsdjvu01. Кроме всего прочего, распространенные форматы лишены средств удаленной навигации, о которых я расскажу чуть позже.

Выходит, от DjVu есть прок только при онлайн-просмотре, верно?

Неверно. Даже если трафик не проблема, рендеринг обгоняет другие форматы и, как правило, требует меньше оперативной памяти.

Похоже, непросто все это осуществить. Как же работает DjVu?

По сути, DyVu – это комбинация трех отдельных технологий анализа и сжатия: DjVuPhoto, DjVuBitonal и DjVuDocument.

Пожалуйста, объясните. Только не все сразу…

DjVuPhoto – это прогрессивная технология сжатия на основе вейвлетов.

А что в ней прогрессивного?

Прогресс в том, что браузер рисует DjVu-изображение непрерывно, с первых секунд загрузки. Вы моментально получаете картинку с низким разрешением, а потом она постепенно обрастает деталями.

То есть можно сразу оценить картинку задолго до полной загрузки?

Именно так. DjVuBitonal – новый тип сжатия, предназначенный для архивирования изображений с большим количеством повторяющихся, практически идентичных, черно-белых значков, известных как «печатные символы». Это может быть любой печатный текст.

Нет слов. Еще сюрпризы от DjVu?

Легко. Третий алгоритм сжатия DjVu создан для того, чтобы сохранять богато иллюстрированные и красочно оформленные журнальные страницы вроде той, на которой протекает наша беседа. DjVuDocument помещает текст и графику, сжатые DjVuBitonal, в один слой, а фотографии и фон, сохраненные DjVuPhoto (возможно, в другом разрешении) – в другой.

А в чем достоинство такого метода?

Любой браузер, распознающий формат DjVu, в первую очередь загрузит текстовый слой. Начинать чтение можно сразу же, пока не подоспели украшения. Разделение изображений и текста упрощает передачу последнего оптическим средствам распознавания образов (OCR).

Но для простых изображений, без текста, выигрыша не будет, так?

Так. Мы уже разобрались, что DjVuDocument создает файлы со слоями для текста и для изображений, и вся спецификация DjVu поддерживает работу с ними. Я не утверждал, что этот формат предпочтителен для чисто текстовых документов, и не говорил, что содержимое текстового слоя обязано быть идентичным таковому в бумажной версии.

Представим себе такую ситуацию: профессор одного из ближневосточных университетов решил опубликовать онлайн древний арабский манускрипт, только что восстановленный «во всей красе». С помощью DjVu можно добавить в файл английский перевод и комментарии. Подробнее об этом можно прочесть здесь: http://djvu.org/resources/embedding_transcripts_in_handwritten_pages.php.

Чудеса. А чем еще DjVu превосходит другие форматы?

Вот, например, проблемы онлайн-просмотра многостраничных документов и пути их решения. Как гласит man-страница DjVu, сканированные версии многостраничных документов в этом формате «не просто коллекция оцифрованных картинок». Чтобы комфортно читать рукописи онлайн (в том числе и при малой скорости подключения), переход от страницы к странице должен быть практически мгновенным, даже если файл еще на полностью загружен!

Ох, верно. Но как DjVu это делает?

С помощью двух форматов хранения многостраничных документов, Indirect (разделяемого) и Bundled (пакетного). Первый генерирует серию файлов, по одному для каждой страницы, и включает в комплект полный гипертекстовый индекс всего документа; благодаря этому возможна загрузка только той части архива, которая вас интересует. Bundled DjVu, напротив, больше подходит для оффлайн-архивирования, так как все упаковывается в единственный файл.

Какое ПО можно раздобыть для создания или отображения DjVu-файлов на Linux?

Стартовая точка для поисков – страница проекта DjVuLibre на http://djvu.sourceforge.net. Это открытая реализация DjVu, которая включает средства просмотра, декодеры, кодеры и вспомогательные утилиты. Например, просмотрщик *DJView4* «понимает» навигацию в документе, проводит поиск и печать, масштабирование, перемещение документа по экрану, создает миниатюры файлов и т.д. Дистрибутив DjVuLibre включает также модули масштабирования и печати для стандартных браузеров Linux, Windows и Mac OS X. Множество документов и программ можно найти на http://djvu.org/resources, а онлайн-сообщество DjVu-пользователей собирается на http://djvu.org/forum.

А как добавить поддержку DjVu к моим собственным программам?

На вышеупомянутой странице Resource есть ссылки и на ресурсы разработчиков, но для Linux-хакера лучше начать со страницы ‘Creating DjVu Documents Linux HOWTO’ (Linux-HOWTO по созданию документов в формате DjVu) на www.howtoforge.com/creating_djvu_documents_on_linux. Там есть еще интересный проект DjVuLibre XML Tools, облегчающий редактирование метаданных, гиперссылок и текста, содержащегося в DjVu-файлах, с помощью XML.

Можно ли провести испытание DjVu?

Конечно. Свободный сервер Any2DjVu (http://any2djvu.djvuzone.org) – то, что вам нужно. Начните с выбора исходного формата: доступны PDF, PostScript, TIFF, JPEG и многие другие. Когда появится HTML-форма, задайте основные параметры преобразования и выгружайте свой файл. Вот и все! LXF

По рецептам д-ра Брауна

Д-р Крис Браун
Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц в этом ему совсем не помогает.

Добротное администрирование систем из причудливых заворотов кишок серверной.

Не пропустите мелкий шрифт

Недавно Microsoft опубликовала свой годовой отчет по форме **10-K** (это документ, который обязаны подавать все публичные компании в США). Пункт 1A, в котором перечислены 20 факторов риска для бизнеса компании, очень интересен. Первая строчка в этом списке гласит «определенные модели ведения бизнеса открытого ПО бросают вызов нашим моделям лицензируемого ПО… Некоторые фирмы конкурируют с нами, используя эту модель… Эти фирмы не несут полных издержек на исследование и разработку ПО. Возможно, некоторые из них используют идеи Microsoft, которые мы предоставляем им бесплатно или за низкие лицензионные отчисления…» (Полная версия документа есть на сайте www.sec.gov).

P-p-p…

С годами Microsoft вобрала огромное количество идей, которые достались ей «за так» – языки C и C++, концепция меню/мыши/иконок, HTML и понятие Всемирной паутины, механизм RPC, используемый DCOM, не говоря уже о TCP/IP! Извините, ну никак не могу относиться к ним с большой симпатией…

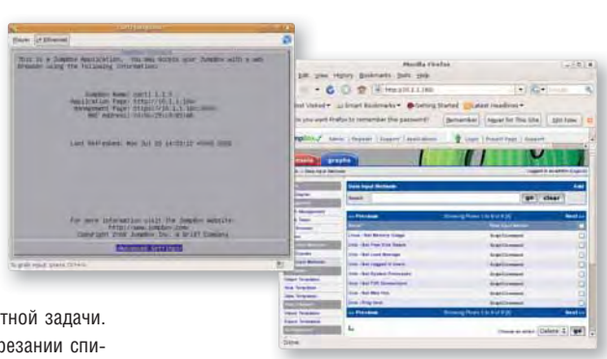
***

И еще кое-что мелким шрифтом, на сей раз в руководстве пользователя нового BT Home Hub. «BT Home Hub содержит код, лицензируемый GPL». Дальше объясняется, как получить исходный код. Я зашел по ссылке на сайт и обнаружил ядро 2.6.8, набор утилит *BusyBox* и *Samba 2.2.12*. Раньше мне как-то не случалось видеть производителя продукта со встроенным Linux, следующего букве GPL – в конце концов, они и двоичных-то файлов отдельно не предоставляют. И зачем ADSL-модему и беспроводному маршрутизатору *Samba*?

Виртуальные приспособления

JumpBox Оцените новые технологии с готовыми настроенными образами виртуальных машин.

Возможность сохранить состояние целого компьютера в одном файле привела к появлению новой технологии создания пакетов программ и их развертывания, которая носит название virtual appliances (виртуальные приспособления, ВП). ВП – это образ виртуальной машины (ВМ), содержащий установленную операционную систему с набором программ, нацеленным на решение конкретной задачи. Настройка обычно заключается только в урезании списка программ до приложений первой необходимости, например, брандмауэра или web-сервера LAMP.



› В окне слева – *JumpBox*, запущенный в *VMware Player*. В окне справа – браузер в основной ОС, который подключается к ВМ.

JumpBox – одна из компаний, которая всерьез работает с этой технологией. Фирма предлагает подписку, по которой ее клиенты получают готовое виртуальное ПО. На сайте JumpBox есть список из 32 пакетов, включающих *SugarCRM*, *Joomla*, *Trac/Subversion*, OpenLDAP, LAMP (Linux, *Apache*, *MySQL*, PHP/ Perl), сервер *MySQL*, *Moodle*, несколько wiki и различные CMS. Базовый вариант подписки стоимостью $200 позволяет запустить один экземпляр каждого пакета ('jumpbox'); в вариантах подписки Plus ($500) и Premium ($1,000) можно запускать 5 и 15 экземпляров соответственно, а также предоставляется поддержка.

Прежде чем негодовать, с чего бы это нужно платить всего лишь за удобную упаковку свободного ПО, вспомните: не приходилось ли вам покупать в супермаркете готовые салаты или баночки с соусом? Думаю, да. Пакеты запускаются в нескольких ВМ: *Parallels* и *VMware Fusion* (в Mac), *Microsoft Virtual Server* и *Virtual PC* (в Windows), *VMware* и *Parallels* (в Linux) и *VMware ESX* или *Virtual Iron* (на «голом железе»). Один пакет можно свободно загрузить, чтобы попробовать в действии. Лично я выбрал *Cacti* (это утилита построения графиков использования сети). Заставить его работать и правда на удивление просто – скачайте пакет, распакуйте и запустите его в *VMware Player*. Результат можно увидеть на экранном снимке выше.

Jumpbox – не один парень на виртуальной деревне. Также можно попробовать www.virtualappliances.net (отсюда можно бесплатно загрузить приспособление для LAMP, LAPP, *Tomcat* и *Cacti*) или на витуальном рынке ВП *VMware* по адресу www.vmware.com/appliances, где кроме всего прочего, можно найти множество образов дистрибутивов Linux. Новичкам в мире виртуализации поможет освоиться наше руководство на стр. 56.

Поджарый или раздутый?

Bonnie++ Управляемый способ измерить производительность файловой системы.

Вы когда-нибудь задумывались над тем, как настройки файловых систем влияют на их производительность? Будет ли быстрее создать их непосредственно в разделе или на логических дисках? Насколько возрастает нагрузка на ресурсы, если файловая система предоставляется хост-ОС в виртуальной среде? Медленнее ли USB-брелки и сетевые устройства хранения данных, чем локальные диски? Стоит ли размещать файл журнала ext3 на отдельном устройстве?

Чтобы ответить на подобные вопросы, нужен способ измерения производительности файловой системы. Одно из решений проблемы – *Bonnie++*. Эту утилиту написал Рассел Кокер [Russell Coker] (см. **www.coker.com.au/bonnie++**); она основана на другой программе, которая называется (угадайте, как?) *Bonnie* и написана Тимом Греем [Tim Gray]. Хотя бинарные файлы *Bonnie++* есть в репозиториях большинства дистрибутивов Linux, можно легко скомпилировать их из исходников. Загрузите tar-архив по ссылке на странице Рассела Кокера. Мы скачали файл **bonnie++-1.03d.tgz** в подходящий каталог. Собрать программу не составило труда:

```
$ wget http://www.coker.com.au/bonnie++/bonnie++-1.03d.tgz
$ tar xzf bonnie++-1.03d.tgz
$ cd bonnie++-1.03d
$ make
```

Как пользоваться Bonnie++

На man-странице перечислены некоторые опции настройки параметров тестирования. В частности, **-d** позволяет указать каталог, для которого будут выполняться все действия; понятно, что нужно указать каталог внутри точки монтирования файловой системы, характеристики которой нужно получить. Также можно задать пользователя, от имени которого будут производиться тесты, и общее количество циклов, которые будут выполнены. Вот пример запуска:

```
$ bonnie++ -m 'USB IDE' -d /media/disk -n 1:1024:0:10 > bonnie.out
Writing with putc()...done
Writing intelligently...done
Rewriting...done
Reading with getc()...done
Reading intelligently...done
start 'em...done...done...done...
Create files in sequential order...done.
Stat files in sequential order...done.
Delete files in sequential order...done.
Create files in random order...done.
Stat files in random order...done.
Delete files in random order...done.
```

Опции команды, которые я в нем использовал, перечислены ниже:

-m: Указать текстовую метку для данного набора результатов. По умолчанию – имя машины.

-d: Указать тестируемый каталог

-n: Указать (a) использовать набор из 1024 файлов для теста по созданию файла, (b) использовать максимальный размер файла 1024 байт, (c) использовать минимальный размер файла 0 байт, и (d) распределить эти файлы по 10 каталогам.

Bonnie++ измеряет производительность трех типов операций: скорости чтения и записи, число обращений к диску в секунду и число операций с метаданными (таких как создание и удаление файла) в секунду. Так как использование файловой системы сильно зависит от компьютера, измерения одного параметра недостаточно. Например, для почтового сервера самый важный параметр – скорость работы с файловыми метаданными. Лучше всего она представлена в параметрах «последовательное создание» ('sequential create') и «случайное создание» ('random create'). С другой стороны, для потокового медиасервера, который в основном читает большие файлы большими порциями, более всего важен параметр «последовательное чтение блоками» ('sequential input/block').

Результаты выводятся в стандартный вывод. Они состоят из общей статистики в текстовом формате, за которой следует одна строка данных, разделенных запятой (CSV-формат), которые можно импортировать в электронную таблицу для сравнения данных двух тестов. Также есть простой фильтр **bon_csv2html**, который преобразует результаты CSV в HTML. В таблице ниже показан результат объединения результатов двух отдельных тестов и передачи тех двух строк в **bon_csv2html**. »

› Вывод *Bonnie++* в формате HTML. Результаты, помеченные '++++', получены для операций, выполнявшихся так быстро, что Bonnie не смогла вычислить осмысленный результат.

Не забудьте Тима Тоди

В соответствии со старым девизом Perl TIMTOWTDI (There Is More Than One Way To Do It – есть не один способ сделать это), имеются и другие средства для снятия характеристик файловых систем. Например, *Postmark* – утилита, которая имитирует загрузку файловой системы, создаваемую файловым сервером. Изначально она была в *NetApps*, а сейчас, похоже, не разрабатывается, хотя и есть в репозиториях некоторых дистрибутивов. Еще одна утилита – *Dbench* от разработчика *Samba* Эндрю Триджелла [Andrew Tridgell], см. **http://samba.org/ftp/tridge/dbench/README**.

Совет дня

Пользуясь какой-нибудь графической утилитой настройки, вы наверняка задумывались: какие же файлы на самом деле изменяются? Вот маленький трюк, который поможет это выяснить. Для начала откройте командную строку и введите:

```
$ touch /tmp/now
```

Теперь запустите графическую утилиту и сделайте все необходимое. Затем вернитесь в командную строку и наберите:

```
find /etc -newer /tmp/now
```

Эта команда покажет, какие файлы в каталоге **/etc** были изменены с момента запуска команды **touch**. Но расширить поиск на всю файловую систему не так просто, потому что будут найдены несколько сотен записей из каталогов **/proc** и **/sys**, которые нам не нужны. Попробуйте такую команду:

```
# find / -newer /tmp/now | grep -v '^/proc' | grep -v '^/sys'
```

Даже после фильтрации командой **grep**, наверное, останется несколько файлов, измененных другими процессами, но сообразить, что это такое, обычно достаточно просто.

		Sequential Output						Sequential Input				Random Seeks			Sequential Create						Random Create					
	Size:Chunk Size	Per Char		Block		Rewrite		Per Char		Block					Num Files	Create		Read		Delete		Create		Read		Delete	
		K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	/sec	% CPU			/sec	% CPU	/sec	% CPU	/sec	% CPU	/sec	% CPU	/sec	% CPU	/sec	% CPU
Without -b	1G	44455	94	53417	12	23631	4	44222	82	55526	3	233.3	0		16	+++++	+++	+++++	+++	+++++	+++	+++++	+++	+++++	+++	+++++	+++
With -b	1G	45308	96	46972	10	23906	4	49085	91	53236	3	225.4	1		16	1019	3	+++++	+++	1152	2	1049	3	+++++	+++	1299	2

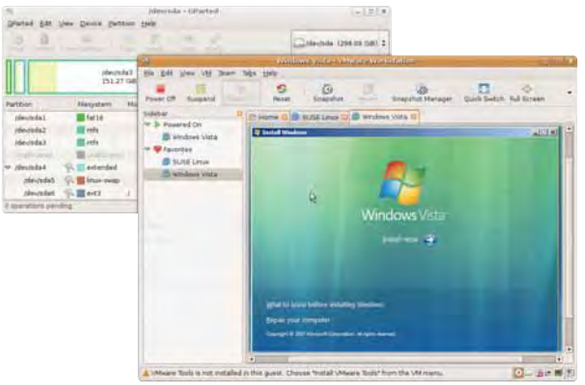
Радости виртуализации

VMware Простое развертывание сервера из образов для виртуальной машины.

Слово «виртуальный» мы в наше время слышим все чаще. Это и виртуальные файловые системы, и виртуальные почтовые каталоги, и виртуальная память, и, восхитительный оксюморон, виртуальная реальность. А теперь можно виртуализовать и сам компьютер, т.е. имитировать компьютерное оборудование, используя программный слой, выполняемый на физической аппаратуре, и установить и запустить операционную систему на этой ВМ. На диаграмме внизу показана архитектура такой системы, хотя и не совсем точная, так как основные части ПО виртуализации на самом деле находится в ядре хост-ОС, а не над ним.

Виртуализация – горячая тема сегодняшнего дня. На виртуальных машинах можно запускать несколько ОС одновременно, бок о бок, на одном физическом компьютере. Они могут предоставить каждой ОС полностью изолированное окружение с ее собственными защищенными ресурсами. Вот несколько причин, по которым используются ВМ:

Образовательное учреждение, где я работаю, интенсивно использует виртуализацию при установке ПО для практических занятий. Виртуализация упрощает создание и распространение образов ПО по учебным центрам со всего мира и предоставляет слой абстракции от различной начинки компьютеров в лабораториях. Если запустить ВМ и включить полноэкранный режим до того, как студенты войдут в класс, то они и не поймут, что работают в виртуальной среде – по крайней мере, пока случайно не нажмут Ctrl+Alt и ВМ не потеряет фокус ввода.



› **Внимательно вглядевшись в экран Vista, вы различите следы *Compiz* из Linux.**

Дадут поиграть

VMware Workstation, наверное, самая известная программа виртуализации; она разрабатывается с 1999 года, и сейчас компания *VMware* предлагает огромный набор соответствующего ПО (www.vmware.com). Некоторые продукты высокого класса довольно дороги, но погрузиться в мир виртуализации можно и бесплатно, скачав бесплатную программу *VMware Server* и воспользовавшись 30-дневным пробным периодом или запуская готовые образы в *VMware Player*. Далее мы рассмотрим процесс инсталляции *VMware Workstation* и создания ВМ, на которую установим Vista.

VMware Workstation можно загрузить в tar-архиве размером 209 МБ с сайта www.vmware.com. Мы скачали файл в **~/downloads** и извлекли его содержимое в этот каталог:

```
$ cd ~/downloads
$ tar xzvf VMware-workstation-6.0.4-93057.i386.tar.gz
```

Появится каталог **vmware-distrib**. Перейдем в него и запустим установочный скрипт:

```
$ cd vmware-distrib
$ sudo ./vmware-install.pl
```

Установщик задаст несколько вопросов насчет того, что и куда копировать, но можно принять все параметры по умолчанию.

> «Установив Vmware, надо создать свою виртуальную машину.»

Настоящие модули ядра

Для *VMware* необходимы три модуля ядра: *vmmon*, *vmblock* и *vmnet*, и их версия должна соответствовать версии ядра. Дистрибутив поставляется с большим набором готовых модулей, и установщик сперва попробует найти среди них подходящие. Не найдя их, он предложит вам компиляцию. По моему опыту, именно здесь и выходит клин. Например, у вас может не быть компилятора *GCC* и других утилит разработки, хотя это-то исправить легко. Для некоторых версий, насколько я знаю, сама компиляция завершалась с ошибками. Однако в данном случае компиляция модулей ядра и оставшаяся часть установки *VMware* прошли без ошибок.

Установив *VMware*, надо создавать свою виртуальную машину. Мастер создания ВМ поможет это сделать. Вот список основных параметров, которые придется задать:

Виртуальный дивиденд

За что мы любим виртуализацию? Давайте посчитаем преимущества…

» Проверка обновлений ПО в изолированной среде перед их установкой в реальной системе.

» Тестирование ПО на различных ОС и конфигурациях, и все на одном компьютере.

» Изучение, проверка и демонстрация многозвенных (клиент-серверных) приложений на одном компьютере.

» Снижение количества необходимых физических серверов и повышение эффективности использования сервера, при сосредоточении нескольких систем на одном компьютере.

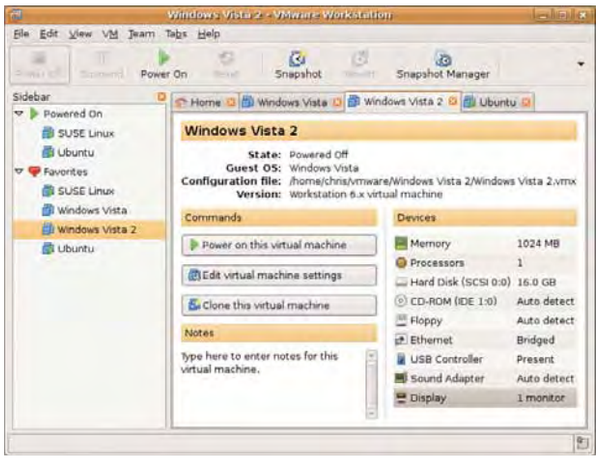
» Виртуальные машины легко переносятся с одной аппаратной платформы на другую, чтобы сбалансировать загрузку или исключить простои сервиса во время сбоев в аппаратной части. В некоторых продуктах ВМ можно перенести даже не прекращая работы. Волшебство!

» Обновление ОС при сохранении возможности запуска важных унаследованных приложений под предыдущей версией системы на ВМ.

» Запуск виртуальных приспособлений – готовых образов для ВМ, предназначенных для выполнения определенной функции или демонстрации определенного продукта.

Все нереально

Поискав слово «виртуальный» в словаре, я нашел следующее определение, применительно к компьютерной области знаний: «Не существующее физически, но видимость существования которого создается ПО». Согласно этому определению, почти все, с чем мы работаем в компьютере, виртуально. В конце концов, даже сняв крышку системного блока, вы не сможете показать ни каталог **/home**, ни письмо от тетушки Мод, ни панель задач, ни процесс init, и вообще ничего, кроме, возможно, орешка, который уронили туда в прошлом году. А вы пробовали когда-нибудь согнать с экрана муху курсором мыши?



Здесь можно выбрать ВМ, просмотреть и изменить ее настройки и выключить или включить ее.

Гостевая операционная система Здесь вы укажете, какую ОС собираетесь установить на ВМ. Этот выбор влияет на некоторые установки по умолчанию, например, объем оперативной памяти и дискового пространства, которые будут выделены для ВМ, а также используется как ее имя по умолчанию.

Имя и Расположение Здесь указывается имя ВМ и каталог, в котором она будет находиться.

Сетевое соединение Есть несколько вариантов настройки сетевого подключения. Сетевой мост (bridged networking) дает виртуальной машине собственный IP-адрес в вашей сети. Хотя ВМ использует Ethernet-соединение хост-компьютера, она появляется в локальной сети как отдельный узел со своими MAC- и IP-адресами. Механизм трансляции адресов NAT (NAT networking) скрывает ВМ от физической сети, но допускает сообщение в обратную сторону точно так же, как шлюз NAT позволяет компьютерам во внутренней корпоративной сети взаимодействовать с Интернетом. Весь трафик в физическую сеть выглядит как исходящий с IP-адреса хост-компьютера. В случае последней конфигурации, «только хост» (host-only networking), ВМ подключается к виртуальному коммутатору Ethernet, позволяющему всем ВМ связываться с хост-компьютером и друг с другом. Физическое соединение между виртуальными машинами и сетью при этом отсутствует.

Размер диска *VMware* предоставляет дисковое пространство хост-компьютера для гостевой операционной системы. Здесь вы задаете размер этого виртуального диска. Для оптимальной производительности все это пространство можно выделить заранее или же дать возможность файлам хост-компьютера разрастаться до заданного максимума по требованию гостевой системы. Также можно разбить файловую систему на несколько файлов по 2 ГБ.

По окончании этого процесса у вас будет два файла: небольшой файл с расширением VMX, содержащий конфигурацию ВМ, и один или несколько гораздо больших файлов VMDK, обеспечивающих дисковое пространство для ВМ. На экранном снимке выше показаны виртуальные ресурсы, выделенные для этой ВМ. Интересны варианты предоставления виртуального диска – можно использовать либо физический CD-привод хост-компьютера, либо ISO-образ, находящийся на нем. Например, если вы загрузили ISO-образ Fedora, можно подключить его напрямую, загрузить с него ВМ и продолжить установку. К тому же образ будет работать быстрее физического диска.

После настройки ВМ необходимо установить на нее операционную систему. Если она инсталлируется с CD или DVD, вставьте его в привод и включите ВМ. Она загрузится с CD, и начнется процедура установки, обычная для выбранной вами ОС.

Программы для виртуализации

Xen Написанный Иэном Праттом [Ian Pratt] из Кембриджа, Xen представляет собой «гипервизор» – относительно тонкий слой, выполняемый на низшем, самом привилегированном уровне. Над этим уровнем выполняется гостевая ОС, которой управляет гипервизор. Первый гость, или, говоря языком Xen, «домен 0», загружается автоматически и получает прямой доступ к «железу». Остальные гости (домены) запускаются из нулевого.

KVM Как можно догадаться по названию, KVM (Kernel Virtual Machine – ВМ ядра) – это свободная инфраструктура для виртуализации, помещенная внутри ядра Linux (доступна начиная с версии 2.6.20). Она поддерживает (а на самом деле требует) процессоры Intel VTX или AMD-V. Я надеюсь рассказать о ней подробно в следующем месяце.

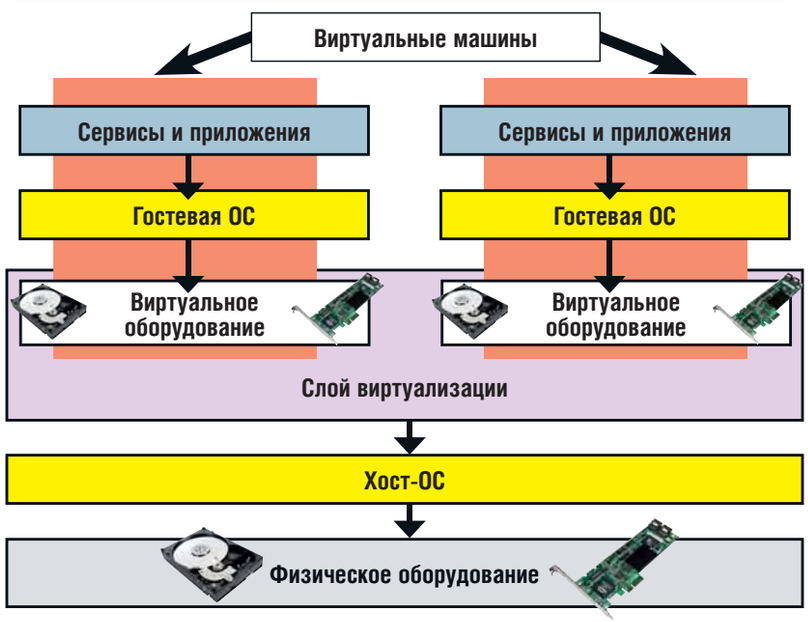
Microsoft Virtual PC Этого продукта, наверное, не должно быть в нашем списке, так как он доступен только в Microsoft Windows и позволяет устанавливать в качестве гостевых ОС только Windows, хотя многие пользователи сообщали, что им удалось заставить работать различные версии Linux.

Parallels Платный ($50) продукт для виртуализации, поддерживающий большой набор хост- и гостевых ОС (Windows и Linux)

VMware Fusion Это продукт для виртуализации от VMware для Mac на процессорах Intel. Я знаю не одного профессионального администратора Linux, который пользуется MacBook Pro, *VMware Fusion* и одним или несколькими дистрибутивами Linux.

VMware Workstation Популярный коммерческий ($190) продукт, поддерживающий Windows и Linux в качестве хост-ОС и большой набор гостевых ОС. О нем мы говорили в этом месяце.

VMware Player Свободная программа, которая может «проигрывать» (но не создавать или редактировать) виртуальные машины, созданные в других продуктах от VMware или загруженные из Интернета.



Виртуализация позволяет запускать несколько гостевых ОС поверх основной.

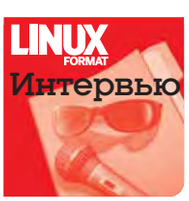
Отпрыгнуть назад

Одна из по-настоящему крутых возможностей *VMware* – возможность создавать снимки состояния ВМ и восстанавливать его позже. Так, если вы установили в Windows-драйвер, который не работает, или думаете, что компьютер подхватил вирус, легко вернуться назад. Каждый снимок включает копию образа всей файловой системы, поэтому большое количество снимков отнимает много места на хост-компьютере.

Ну, мы виртуально закончили. Собственно говоря, мы и вправду закончили. В следующем месяце мы подробно рассмотрим другую программу для виртуализации – *KVM*. Увидимся! LXF

Повелитель машин

Linux, конечно, не NetBSD, но тоже может работать на всем, у чего есть процессор. Например, на самодвижущемся роботе, как выяснил **Антон Борисов**.



Если вы – поклонник изобретательства и конструирования, то наверняка в курсе, что передвижные установки в виде радиоуправляемых машин делаются давно и вполне успешно. Взять хотя бы конкурс под названием Grand Challenge, спонсируемый DARPA. Ежегодно несколько научных заведений выпускают на турнир свои радиоуправляемые установки. Цель машин – автоматически, насколько это вообще возможно для искусственного интеллекта на сегодняшний день, пройти всю трассу турнира. Люди, выступают лишь в качестве зрителей. Шасси, на которых сконструированы машины, представлены разными автомобильными компаниями и несут на себе различное оборудование – от сенсоров движения до солнечных батарей.

Впрочем, для домашних экспериментов нет смысла изобретать велосипед, в смысле, подобного монстра, в виде настоящей автомашины: можно сконструировать нечто подобное из доступных компонентов, как сделал **Джон Гэллоуэй** [John Galloway], основатель проекта Toasterfish (www.toasterfish.com).

LXF: Джон, как вы пришли к мысли сконструировать вашего робота?

Джон Гэллоуэй (ДГ): Так получилось, что однажды я купил пару web-камер от Apple – Apple iSight – и воспользовался ими для организации видеоконференции на двух моих домашних компьютерах. Через некоторое время мне в голову пришла мысль, что если одну из этих web-камер установить на некое подобие робота, то может получиться отличный эксперимент. Представьте: вы сидите дома, а путешествующий робот передает вам изображение того, что он видит. Я могу услышать и увидеть то, что видит робот, а он, в свою очередь, передаст мои изображение и голос тем людям, что находятся поблизости. Эта идея захватила меня, и я стал думать, как реализовать ее на практике.

Поначалу кажется, что стоит купить роботизированную платформу и установить на ней пакет программ, которые организуют телеприсутствие. Что может быть проще? Но все оказалось немного сложнее. Такого проекта, что совмещает как аппаратную, так и программную части, не существует. Решения, что есть на рынке, начинаются от суммы в $3000 – они предназначены в основном для образовательных целей. Для более серьезных академических исследований суммы, конечно же, еще серьезнее. Понятно, что не каждый может позволить себе удовольствие потратить такую сумму на «игрушечный» проект. Более того, оказалось, что даже те дорогие продукты не предназначены для какой-либо сложной экспериментальной базы. В их дизайн не была заложена возможность масштабирования, т.е. получалось, что у вас есть конструктор, из которого можно собрать только один-единственный вариант. Поэтому я и решил произвести свой собственный конструктор.

Отправной точкой для начала экспериментов стали радиоуправляемые машины. Слегка продвинувшись в данном направлении, я выяснил, что оно отнюдь не является целиной, и многие наработали определенный опыт в этой сфере: например, проект MARS в университете Пенсильвании [1]. Они начинали с радиоуправляемой игрушки Cloudbuster, что производит компания Tamiya. Я купил пару таких же моделей на аукционе E-bay и приступил к изучению их схематики. Вскоре выяснилось, что они выглядят здорово только внешне, а для применения в роботе требуется серьезная переделка.

К счастью, индустрия радиоуправляемых моделей полностью имитирует рынок полноформатных автомобилей – в том плане, что в ней есть конкуренция. Кроме основных игроков, таких как Tamiya, есть и серия мелких производителей, которые занимаются «тюнингом», если можно так выразиться. Одна из них – ThunderTech Racing [2]. В разговоре с Дарреном (как выяснилось, эта компания состоит, по сути, из одного человека) по поводу моего проекта, он упомянул что недавно общался с ребятами из западного кампуса университета Карнеги-Меллона относительно деталей аналогичного проекта, и пригласил меня в университет на первую презентацию.

В сотрудничестве с университетом Даррен изготовил несколько платформ, которые состояли из компонентов Cloudbuster, таких как колеса, покрышки, мотоподвес, тормозные механизмы. Однако, в тоже время, на этой платформе можно было разместить и материнскую плату формата mini-ITX. Окончательная версия этой платформы используется университетом в RoboCamp [3] – событии, устраиваемом университетом для старших школьников, интересующихся

› **Прототип робота.**

робототехникой. Я попросил Даррена внести небольшие изменения в разработанную платформу, что позволило бы мне прикрепить LCD-экран и web-камеру.

LXF: Как я понимаю, под впечатлением от большого автопрома вы и назвали своего робота?

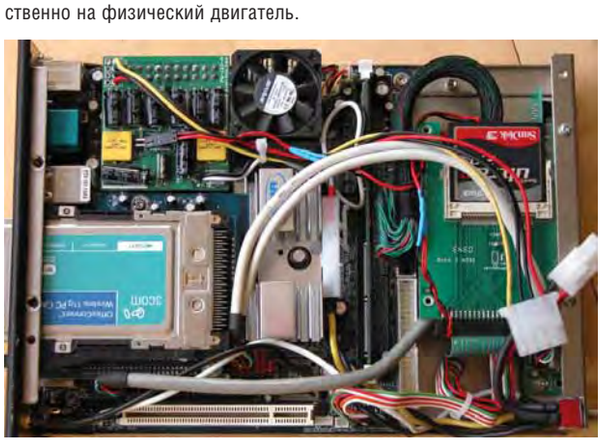
ДГ: Да. Сначала я назвал его Monsterbot, т.к. он выглядел как маленький Monster Truck – это такие огромные машины, которые вы можете видеть на специализированных автошоу. Там они перепрыгивают через легковушки и крушат их. Однако затем у компании Monster Cable Products, что организовывает эти шоу, возникли претензии к имени «Monster», поэтому я выбрал «Toasterfish». У него нет никакого определенного значения; меня навел на него знакомый, когда мы занимались дайвингом и он пытался описать мне рыбу, похожую на тостер (на самом деле он имел в виду кузовка [4], очертания которого достаточно угловаты, как и у моего робота).

LXF: Из каких компонентов состоит робот?

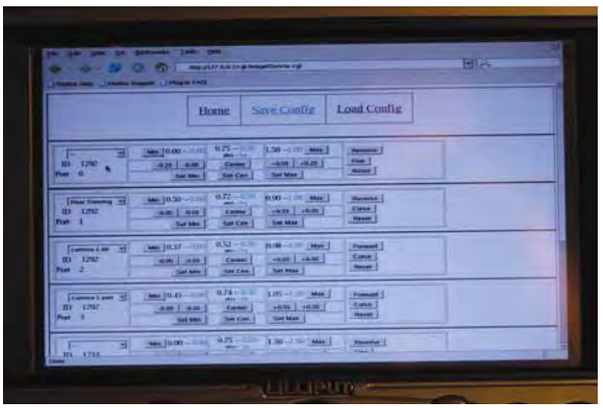
ДГ: Центральная мозговая система устройства – материнская плата VIA EPIA MII-10000 формата mini-ITX (в разработке Карнеги-Меллон использовалась другая модель, но того же производителя). Управление сервоприводами, контролирующими колеса, поворотом камеры и скоростью движения осуществляется через дополнительную плату Phidget [5], подключенную к USB-порту VIA. Вторая плата, что входит в комплект – это карта 802.11 WiFi в формате PCMCIA. Она обеспечивает сетевое управление роботом и передачу видеосигнала. Плюс пара конвертеров постоянного тока DC-DC, которые преобразуют напряжение аккумулятора в требуемое материнской плате и сервомеханизмам. Управление передними и задними колесами, а также вращение камеры происходит через радиоканал. Сигнал поступает на приемный тракт WiFi-карты, затем происходит его обработка внутри «мозга» с помощью программного обеспечения, и через USB-разъем подается нужный импульс на плату расширения (в моем случае – Phidget), а уж с нее нужные команды поступают на двигатели. Таким вот образом осуществляется движение робота.

LXF: Можно ли рассказать более подробно о программной составляющей робота?

ДГ: Я использовал урезанную версию Linux от Red Hat, на основе ядра 2.6. Вся операционная система уместилась на 2-ГБ карточку CompactFlash; она подключается к материнской плате через переходник CF–IDE. Разработка Университета Карнеги-Меллона (она отлична от моей) основана на версии Linux от MathLab. Все необходимое ПО было написано мною – в общем-то, ничего сложного. Интерфейс управления серво-механизмами – разделяемая память. Каждый виртуальный сервопривод контролируется демоном, который пишет в определенный участок памяти нужное содержимое. И затем сигнал, через описанную выше схему и USB-разъем, приходит непосредственно на физический двигатель.



› Операционная система робота загружается с карты CompactFlash, команды телеметрии передаются по Wi-Fi.



› Apache и cgi-скрипты — «рабочие лошадки» программы управления роботом.

Затем я подготовил web-интерфейс, чтобы выставлять нужные значения на HTML-странице и с помощью CGI-скриптов передавать их демону сервоприводов. Web-интерфейс доступен по WiFi-сети, поэтому у себя на рабочей машине в браузере я выбираю значения, а робот изменяет траекторию своего движения. Должен признать, что управление через интерфейс во время движения робота немного замедленное, т.к. ширину канала забивает передача видеоизображения. Поэтому я сделал альтернативный вариант, с намного меньшей задержкой, используя RPC (Remote Procedure Call, удаленный вызов процедур).

Всего этого оказалось достаточно, чтобы начать предметный разговор о телеприсутствии робота, что, в общем-то, и было моей конечной целью в разработке. Пока робот, правда, не умеет самостоятельно передвигаться, но это вопрос разработки ПО с искусственным интеллектом и распознавания образов. Разработанный вариант – хорошая база для отладки систем контроля, которая позволит создать функции автономного перемещения значительно быстрее.

LXF: А почему вы выбрали Linux, а, скажем, не QNX, VxWorks или какую-либо другую встраиваемую систему?

ДГ: В основном потому, что Linux – открытая и свободная система. И является основным инструментом для многих исследователей, «хакеров» и энтузиастов своего дела. У Linux существует ряд проблем для данного применения – в первую очередь, отсутствие настоящего «реального времени», т.е. гарантий, что любая задача при необходимости будет обслужена в течение 10 миллисекунд. На настоящий момент данное условие жестко гарантировать нельзя. Однако, благодаря открытости Linux, эти вопросы разрешатся, если будет достаточная заинтересованность в данном направлении.

Не последнюю роль играет и стоимость операционной системы. Т.к. целевая аудитория, как я уже упомянул, это исследовательские, научные и учебные кадры, то использование лицензируемых систем, на мой взгляд, кажется совсем не оптимальным решением.

LXF: Каковы, на ваш взгляд, возможные сферы применения данного робота?

ДГ: Применение объектов телеприсутствия может быть очень разнообразным. В качестве примера – представьте, что робот расположен рядом со школой вашего ребенка. Вы можете присоединиться к нему из своего рабочего офиса, с помощью телеметрии управлять его движением, найти малыша и передать ему привет. Или такой пример. Вы находитесь в отпуске и хотите проконтролировать ситуацию дома: например, как себя ведут домашние питомцы. В целом, это тоже возможно.

Однако цель, которую я себе поставил, начав разрабатывать робота – это привлечение большего внимания к данной области. Проблемы, которые возникают при разработке мобильных сервисов, почти всегда касаются программ, а не механики или электроники. Чтобы привлечь максимальное внимание к робототехнике, необходимо в то же время отвлечь внимание программистов от ненужной суеты с механикой. Посмотрите: чтобы запустить роботизированную

платформу, программистам придется для начала изготовить опытный образец, и лишь затем приступить непосредственно к программированию – навигации, управлению задачами, распознаванию образов и т.д. Данное условие сильно сокращает число желающих поработать в области роботостроения. Так что главным условием было предложить готовую платформу, где технические вопросы – например, управление скоростью передвижения – давно решены, и можно полностью сосредоточиться на чисто программной части.

Мой созданный прототип нельзя назвать полностью расширяемым. Для создания объектов телеприсутствия, а именно этой целью я изначально задавался, он подходит замечательно. Но чтобы быть действительно универсальной платформой для разработки, нужно представить новое шасси, новую электронику и новое ПО. К сожалению, привлечь внешнее финансирование мне пока не удалось, поэтому проектом занимаюсь в свободное время, а на следующий прототип явно придется потратить больше времени.

LXF: Участвовал ли ваш робот в ралли с искусственным интеллектом?

ДГ: Вы, наверное, имеете в виду RoboMagellan – микроверсию ралли Grand Challenge, которое устраивает агентство DARPA? К сожалению, нет. В ралли RoboMagellan роботам раздаются координаты конусов, расположенных в 20 метрах друг от друга, и задача участников – объехать эти препятствия в автоматическом режиме. Как я уже упомянул ранее, ПО для автономного поведения моего робота только предстоит создать. По всей видимости, эту задачу придется реализовывать в следующей версии прототипа.

LXF: Джон, насколько я вижу, вы также используете в своем роботе GPS-модуль. Насколько он точен в получении координат со спутника? И еще вопрос по поводу телеприсутствия: какова полоса пропускания для передачи изображения с камеры по Wi-Fi? Потому что сразу возникает вопрос: а удастся ли создать группу из, скажем, 5 или 10 монстр-ботов, объединенных медленной радиосвязью? Они могли бы обмениваться своими координатами и «знать», где именно находится ближайший сосед — этакая команда разведки ExploreTeam в пустыне. Каковы ваши идеи на этот счет?

ДГ: Применение GPS очень заманчиво, но в тоже время у этой системы есть определенные ограничения. Не впадая в подробности, скажу, что с одного модуля GPS можно на местности получить точность до 7 м. При использовании 2 и более модулей, разнесенных в пространстве, эта точность повышается до 2 метров. При передвижении точность можно еще повысить. В теории, точность на местности можно повысить вплоть до 2 см, но для этого придется использовать направленную спутниковую антенну диаметром в 20 см и стоимостью от 10000 долларов США. Такие антенны обычно используют для создания точных дренажных систем в сельском хозяйстве.

Правда, я видел и очень точные измерения, где с помощью дифференцирования получали отклонение в 1 метр с использованием обычных GPS-датчиков. Так что факторов, которые влияют на GPS, очень и очень много. Со временем результаты измерений улучшатся – с появлением новой европейской системы и следующего поколения GPS-спутников. Правда, это случится не ранее следующих 5–10 лет.

Для видео требуется удивительно малая полоса пропускания. С использованием современных кодеков полосы в 200 Кбит/с хватает за глаза, в то время как стандарт 802.11 позволяет передавать даже 10 Мбит/с (для стандарта 802.11g от 30 Мбит/с и выше). Так что для «общения» роботов друг с другом вполне хватит. Сегодняшний вариант робота передает видео в виде потока MPEG, но я считаю это не совсем оптимальным решением и думаю сделать передачу видео на основе JPEG2000. К тому же, этот кодек менее требователен к ресурсам процессора.

Исследования в области обмена информацией для построения объемной карты местности, собранной из нескольких независимых источников, на данный момент являются очень актуальными. И скорее всего, в ближайшее время появится много продуктов, где будут применяться данные технологии. Стоит отдавать себе отчет в том, что синтезированное изображение получается более четким, нежели полученное от любого единичного робота. С помощью протоколов и ПО для создания ad-hoc сетей, боты могут поддерживать соединения между собой даже при отсутствии сообщения с некоторыми соседями. Беспроводные карты не могут похвастаться большой дальностью связи. Чтобы обойти это ограничение, используют такую комбинацию: направленные антенны Яги – для дальней связи; ненаправленные, прикрепленные рядом с видеокамерой на крыше – для ближней. Боты изучают местность, затем через определенный момент времени останавливаются, «находят» друг друга, обмениваются информацией и путешествуют дальше. Применений – бесконечное количество.

LXF: Вы упомянули, что следующий прототип будет немного отличаться от сегодняшней версии. Каковы основные различия?

ДГ: Сконструированная версия робота – на самом деле эксперимент, имеющий целью убедиться, что выбранный форм-фактор является правильным. И он полностью себя оправдал. Давайте посмотрим еще раз, из чего состоит робот: передаточный механизм взят от Tamiya CloudBuster, шасси было изготовлено по спецдизайну компанией ThunderTech. Сервоприводы управляются через контроллер Phidget. Используется камера Apple iSight. Робот питается от двух Ni-MH батарей емкостью 3000 мА-ч – 4 секции предназначены для питания электроники и 2 секции – для питания двигателей. Скорость двигателей изменяется контроллером SuperRooster.

Следующий прототип видоизменится. Во-первых, все четыре колеса будут управляться независимо друг от друга, в том числе можно будет выставлять разную скорость вращения колес. Во-вторых, сам корпус материнской платы будет немного побольше, чтобы можно было вставить PCI-карту для управления всеми сенсорами и электромоторами. Шасси будет нести уже закрытый компьютерный корпус, а не полностью открытый, как сделано сейчас. Батареи заменятся на специализированные литий-ионные, с вдвое большей емкостью.

Конечно, для российского рынка трудно равняться на те возможности, что существуют, например, в той же Кремниевой долине в Калифорнии или индустриальном районе в одной из провинций Китая, где не требуется прикладывать определенных усилий, чтобы найти именно то, что нужно, и главное, потом доставить это на дом. С другой стороны, когда вы точно знаете те устройства, что нужны вам, и где их найти, то считайте, что половина дела уже сделана. Не думаю, что доставка из Германии, а именно там я нашел представителя от Phidget, будет стоить астрономических денег. Да и продукты компании Tamiya в РФ тоже доступны, например, здесь [6–8]... Ну, это если вы совсем уж «бородатый хакер» и хотите знать все, вплоть для мельчайших деталей. Для тех же, кому больше по душе программирование и некогда заниматься логистикой компонентов, могу посоветовать обратиться к автору оригинального дизайна. В любом случае, в выигрыше остаются все. Если есть аппаратный объект моделирования, то ждать новых идей из академических кругов долго не приходится. Стоит только оглядеться. LXF

Полезные ссылки

1. Проект MARS – www.cis.upenn.edu/mars
2. ThunderTech Racing – www.thundertechracing.com
3. RoboCamp – west.cmu.edu/west_connect/robocamp
4. Кузовок, он же рыба-тостер – www.freshmarine.com/longhorn-cowfish.html
5. Phidgets – www.phidgets.com
6. Модели Tamiya в России – xmodels.ru/catalog/87/3691
7. Еще модели Tamiya в России – smartmodels.ru/product_info.php?products_id=193
8. И снова Tamiya – superhobby.ru/index.php?section=news&id=106

LINUX FORMAT

Учебники

Наши эксперты помогут вам с любым приложением Linux

ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

Ричард Столлмен — нобелевский лауреат?

Добро, которое вы сотворили сегодня, люди позабудут назавтра; Все равно творите добро.

Мать Тереза.

С премиями по физике, химии, физиологии и медицине заведомо ничего не светит. По литературе, при всем моем уважении к его «Праву читать», тоже вряд ли что-то выйдет. Остается только Нобелевская премия мира. Согласно завещанию Альфреда Нобеля, она должна вручаться «лицу, которое внесет наибольший вклад в дело укрепления содружества наций, в ликвидацию или снижение напряженности противостояния вооруженных сил, а также в организацию или содействие проведению конгрессов миролюбивых сил».

«Снижение напряженности противостояния вооруженных сил», конечно, не очень подходит, но почти половина Нобелевских премий мира после Второй мировой войны также вручалась без оглядки на этот пункт. Укрепляет ли дело Столлмана содружество наций? Безусловно, так как дело его не имеет границ. Защищает ли философия открытого ПО права людей? Конечно, так как эта философия позволяет объединиться и дать отпор сильным и жадным. Содействует ли он проведению собраний миролюбивых сил? Мы очевидно миролюбивы. Свободное ПО может создаваться только в мирное время, так как война – это естественный враг свободы.

Одним словом, Ричард Мэттью Столлмен – вылитый кандидат в лауреаты. Одна загвоздка: его кто-то должен выдвинуть. Вот с этим проблемы, так как вряд ли, например, Международный арбитражный суд в Гааге, имеющий такое право, сегодня проголосует за него. Но я надеюсь, что мир со временем примет этого кандидата, как принял ту же Мать Терезу.

E.M.Baldin@inp.nsk.su

В этом выпуске...

62 Украсим Gnome
Пользователи этого рабочего стола могут немного завидовать KDE 4 с ее плазмоидами. **Энди Ченнел** покажет, что *GDesklets* ничуть не хуже и объяснит, как настроить двойную загрузку Linux и Vista.

66 Гуру выделений GIM
Мы все используем эти инструменты, но редко выходим за пределы простых кругов и квадратов. **Майкл Дж. Хэммел** обучит вас связыванию областей, лассо и быстрым маскам. Выделиссимо!

70 Овладеваем мультимедиа
Не отставайте от прогресса: разберитесь с новейшими мультимедиа-приложениями. **Марко Фиоретти** намекнет, что заслуживает внимания.

74 Синхронизируем календарь
Досадное недоразумение №21: на вашем ПК, ноутбуке и сотовом телефоне – везде свои календари. Но не беспокойтесь, **Саймон Пиксток** избавит вас от этой головной боли.

78 Следим за сетью
Nagios – чрезвычайно мощный инструмент для мониторинга ценных узлов вашей сети. **Джульетта Кемп** поможет установить его и обрисует основные возможности.

82 CMake: допглавы
Все, что мы хотели рассказать, но не смогли упихать в учебники прошлых номеров: **Андрей Боровский** рассматривает расширения *CMake* и создание дистрибутивов.

85 Перевод по правилам
Отечественный журнал не может игнорировать проблему интернационализации приложений – **Андрей Боровский** покажет, как сделать это применительно к KDE 4.

88 гРафические интеRфейсы
Дайте волю своей мыши – **Антон Коробейников** и **Евгений Балдин** представляют десятку самых простых (но необязательно самых правильных) способов работы в R.

92 Вычисления в стиле «грид»
Большой адронный коллайдер остановился до весны, но его дело живет! **Андрей Шевель** расшифрует модное ныне слово «грид» на примере пакета *gLite*.

Совет месяца: Блокируем ботов

Устали наблюдать в системных журналах сотни (и даже тысячи) записей о провалившихся попытках входа через SSH, создаваемых тупыми ботами, пытающимися прорваться на ваш сервер? Пока они не угадают пароль, вреда не много, но на нервы действует. К счастью, есть масса способов от них избавиться. Самый лучший (при условии, что вы никогда не подключаетесь к своему серверу извне) – просто закрыть порт 22 на маршрутизаторе; тут уж никто не пройдет. Альтернатива – программы вроде *Fail2ban* (http://fail2ban.sourceforge.net) или *DenyHosts* (www.denyhosts.net). Они просматривают журналы в поисках повторяющихся неудачных попыток входа в систему с одного IP и временно блокируют его средствами установленного у вас брандмауэра.

Третий вариант поразительно прост. Взломщики SSH обычно ожидают увидеть данную службу на порту 22 – измените этот номер на случайное большое число, и они «отвалятся». Отредактируйте директиву **Listen в /etc/ssh/sshd_con g** примерно так

```
Listen 31337
```

а затем перезапустите *sshd*. Единственный недостаток данного подхода в том, что вам придется постоянно указывать выбранный номер порта в команде *ssh*. Чтобы избежать подобных неудобств, используйте псевдоним оболочки:

```
alias myssh ssh -p 31337
```

GDesklets: Ау,

Что-то прекрасно на вид, а что-то умно. **Энди Ченнел** нашел вещи, сочетающие оба качества: родные для Gnome виджеты *GDesklets*.

Наш эксперт

Энди Ченнел
Энди делает свои первые шаги в Linux уже шесть лет, а технологиями интересуется со времен Dragon 32.

Если вы использовали или видели Mac, то могли обратить внимание на панель Dashboard – быстрый, приятный способ отображения классных гаджетов. Хотя Apple Dashboard, наверное, наиболее узнаваемая форма настольных гаджетов, для Linux имеется ряд подобных решений, включая виджеты Google (мы рассматривали их в LXF109), Opera и KDE'шные плазмоиды. На данном уроке мы намереваемся установить *GDesklets*, родную для Gnome систему виджетов, и слегка ее заселить. Мы также кратко коснемся альтернативной системы под названием Screenlets, которая, согласно некоторым отчетам, работает лучше, если вы используете все крутые эффекты *Compiz*, предоставляемые теперь современными дистрибутивами. Однако мы не отметили проблем использования *GDesklets* с *Compiz* в Ubuntu. Мы протестировали его, используя OpenSUSE 11 и Ubuntu 8.04 (Hardy Heron). В первом пакет *GDesklets* доступен через Yast; пользователи других систем должны проверить свои менеджеры пакетов или взять двоичные файлы с сайта проекта (**www.gdesklets.org**). Пользователи Ubuntu могут выполнить

```
sudo apt-get install gdesklets gdesklets-data
```

В SUSE мы открыли Computer > Install Software [Компьютер > Установить программы] и выполнили поиск *GDesklets*. Отметив пакеты, мы выбрали Install [Установить], а затем Apply [Применить] и подождали завершения загрузки 4,6 МБ. По завершении всех операций нажимаем Computer > More Applications [Компьютер > Дополнительные приложения]. *GDesklets* должен появиться в разделе New Application [Новое приложение], а когда он уже не будет рассматриваться как новинка, то в Utilities [Утилиты]. В Ubuntu это будет Applications > Accessories [Приложения > Вспомогательные].

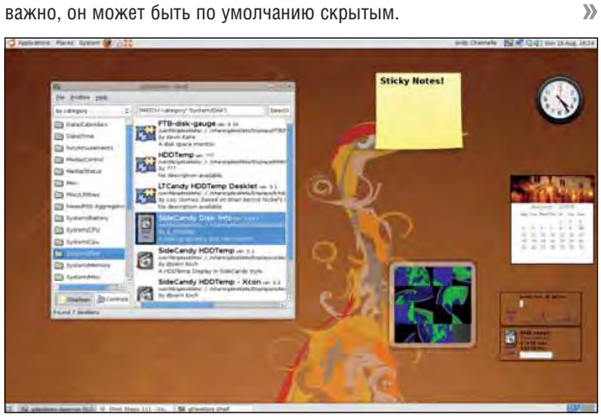
Последняя версия *GDesklets* отличается тем, что разработчики называют Плавающий режим [Float mode]: он позволяет настроить комбинацию клавиш (по умолчанию это Shift+F12) для вывода десклета поверх обычного рабочего стола. Для пользователей старых версий *X.org* в использовании этих функций имеется несколько нюансов, то есть десклеты выглядят не так хорошо, как должны. При щелчке на иконке *GDesklets* будет запущено приложение *GDesklets Shell*, используемое для добавления и удаления десклетов, и в системном лотке появится иконка *GDesklets* – в SUSE она размещается в правом нижнем углу, в Ubuntu в правом верхнем.

GDesklets Shell выглядит как обычное настольное приложение с меню вверху, панелью с набором папок слева и большой областью, отображающей список доступных десклетов (в зависимости от выбранной папки) справа. Десклеты объединены по сфере применения: например, раздел под названием Дата/Время содержит часы, а в Развлечения попадают всякие пустячки.

Приступим

На рабочий стол десклет добавляется двойным щелчком по нему. При этом десклет «прилипнет» к мыши, и его можно будет «уронить» в любом месте экрана. Поскольку эти апплеты разработаны как статичные, их перемещение – нечто большее, чем просто «перетащить и бросить». Для вывода слоя десклетов на передний план нажмите Shift+F12 – они будут размещены поверх всех текущих окон; а затем щелкните правой кнопкой мыши на желаемом десклете, выберите Move Desklet [Переместить десклет], чтобы загрузить его вновь, и перенесите его в любое место рабочего стола. Используйте Shift+F12 опять, чтобы перенести слой десклетов в самый низ.

Screenlets делает то же самое, но предлагает более тесную интеграцию с системой оформления *Compiz Fusion*. Последняя версия, 0.36, доступна на **www.screenlets.org**. Одним из главных отличий является то, что настроить *Screenlets* в качестве слоя рабочего стола для работы с *Compiz* намного проще, то есть отображение виджета может быть связано с комбинацией клавиш в самом *Compiz* и, что, наверное, более важно, он может быть по умолчанию скрытым.



Десклеты, охватывающие производительность, информацию и напоминалки, добавляются на рабочий стол посредством *GDesklets Shell*.

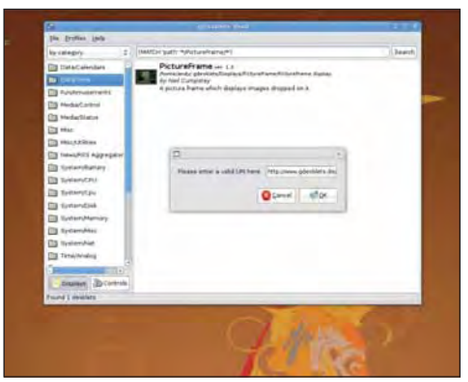
Месяц назад Мы сберегли массу времени и трафика при помощи BitTorrent и Vuze.

красавчики!

Шаг за шагом: Разбираемся с GDesklets

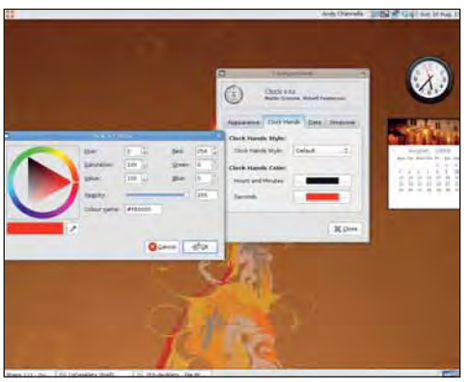
1 Загрузите…

Богатый выбор десклетов имеется на www.gdesklets.de. Это обыкновенные tar-архивы, и после их загрузки ваш дистрибутив, вероятно, предложит распаковать их. Не делайте этого; взамен сохраните их где-нибудь и в *GDesklets Shell* выполните File > Install Package… [Файл > Установить пакет...], а затем укажите скачанный архив. Новый десклет добавится в список, и его можно будет выбирать обычным способом.



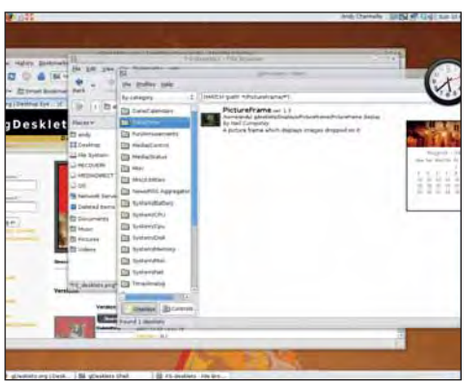
2 … или установите из web

Доступная альтернатива – установка нового десклета без предварительной загрузки. Вместо щелчка на ссылке сайта Download [Загрузить], щелкните правой кнопкой мыши и выберите Copy Link Location [Копировать ссылку]. Вернитесь в Shell, выполните File > Install Remote Package [Файл > Установить удаленный пакет] и вставьте URL в соответствующую строку. Нажмите OK, чтобы добавить его.



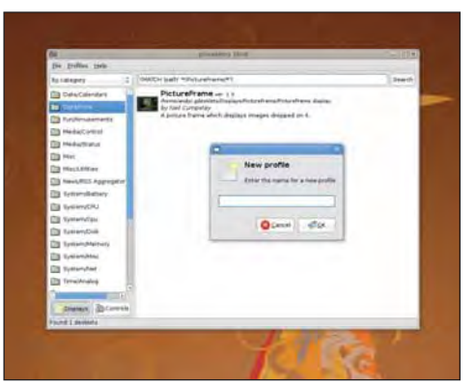
3 Настройка

Хотя *GDesklets Shell* конфигурирует всю систему десклетов, каждый отдельные апплет также является настраиваемым. Например, для погоды можно указать часовой пояс и место расположения, а для RSS-читалки можно определить адрес ленты новостей. Для доступа к этим опциям щелкните правой кнопкой мыши на десклете и выберите Configure Desklet [Настроить десклет].



4 Вид

По умолчанию десклеты находятся у нижней кромки экрана, так что остальные объекты закрывают их. Чтобы увидеть десклеты, нажмите Shift+F12 – тогда они всплывут на самый верх. Для редактирования десклетов не обязательно делать их «всплывающими», но иногда это помогает.



5 Профили

Одной из действительно хороших функций являются Профили, позволяющие определять различные наборы виджетов, скажем, для дома и работы. Выбрав Profiles > Profile Name [Профили > Имя профиля], вы увидите только десклеты, соответствующие этому режиму работы. Создайте новый профиль при помощи Profiles > New Profile [Профили > Новый профиль] и добавьте соответствующие десклеты. Профили запоминают не только отображаемые десклеты, но также их расположение на рабочем столе.

6 Настройка приложения

Для доступа к приложению настройки *GDesklets*, откройте терминал (или нажмите Alt+F2 для открытия диалога запуска) и введите **gdesklets configuration**. Здесь есть опции для отображения иконки в системном лотке и изменения разрешения экрана, а также настройка комбинации клавиш, активирующей плавающий режим. По умолчанию это Alt+F12, что может быть изменено путем щелчка на кнопке Float Mode и нажатия желаемых клавиш.

Двойная загрузка:

Не готовы окончательно попрощаться с Microsoft? **Энди Ченнел** покажет, как заставить Linux и Vista работать бок о бок на одной машине.

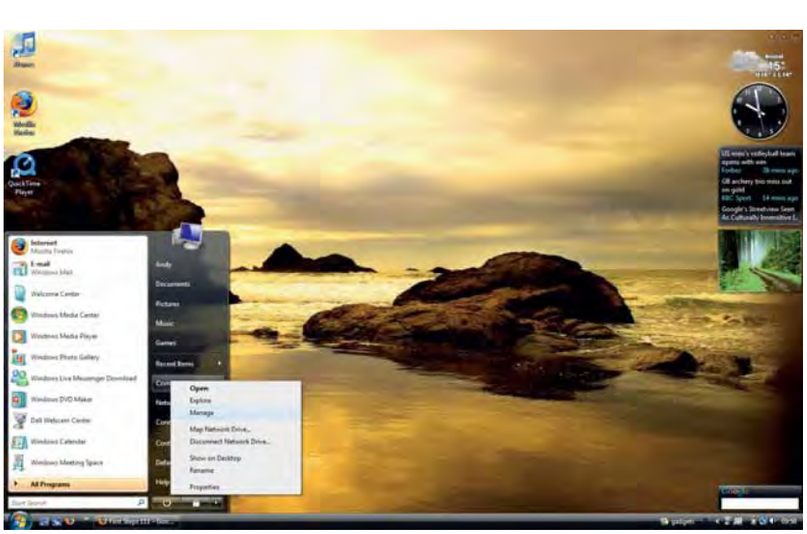
Скорая помощь

Выполняя установку на ноутбуке, позаботьтесь, чтобы он был подключен к сети – батарея может сесть на половине процесса.

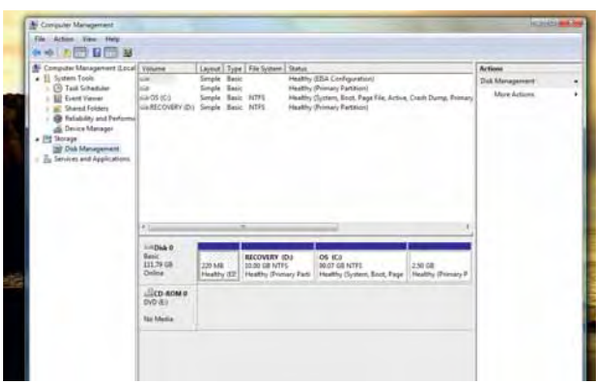
Если вы только что купили ПК с сияющим голубым логотипом Windows Vista, то этот учебник – для вас. Мы разобьем жесткий диск, уже содержащий Vista, и обновим его установкой свежего Ubuntu, так что компьютер будет иметь двойную загрузку. Двойная загрузка – это процесс, в котором две (или более) операционных системы могут быть добавлены на одну машину. При включении машины пользователь просто выбирает, какую ОС загружать, и она стартует как обычно. Хотя обе ОС размещаются на одном жестком диске, компьютер об этом не подозревает: одним из фундаментальных аспектов двойной загрузки является разбиение винчестера, что приводит созданию двух (и более) «дисков» из одного. Часто можно получить доступ к данным, скажем, к разделу Windows из Linux (или наоборот), то есть вы можете хранить все ваши MP3-файлы в одном разделе и иметь к ним доступ вне зависимости от работающей в данный момент ОС.

Прежде всего нам следует очистить на жестком диске пространство для Linux. Предыдущие версии Windows (98, XP) не содержали инструментов разбиения дисков [во многих дистрибутивах Windows присутствует утилита командной строки *fdisk*, предназначенная именно для этого. Кроме того, она есть на любой загрузочной дискете восстановления Winodws, – *прим. пер.*], поэтому установщики Linux весьма подналторели в их «нарезании». Однако Vista имеет собственный инструмент разбиения, и он, как нашли некоторые пользователи, более устойчив в начале процесса. Перед тем, как приступить к выполнению любых операций вроде разбиения или освобождения места, следует сделать две вещи. Во-первых, определить метод восстановления любой ОС, которую вы можете случайно повредить, и во-вторых, сделать резервное копирование всех важных данных, потеря которых нежелательна. Держа в уме эти предупреждения, давайте приступим.

Загрузите Vista и, в меню Пуск, щелкните правой кнопкой мыши на ярлыке Мой компьютер и выберите Управление. Запустится главное приложение администрирования системы; в нем вы сможете воспользоваться древовидным списком в левой панели для выбора Устройства хранения > Управление дисками. После короткой паузы главная панель



› Утилита управления дисками доступна из меню Пуск



› Это стандартная схема разделов на машине Dell только что из магазина. Интересующая нас часть обозначена OS (C:).

этого окна отобразит схему разделов вашего компьютера, где каждый из них будет представлен отдельным блоком, который можно выбрать.

Выполните правый щелчок на разделе с меткой **C:**. Скорее всего, это будет самый большой раздел в системе, и иногда он может помечаться как **OS**. В появившемся меню выберите Shrink Volume [Уменьшить раздел]. Имеется пара ограничений, о которых вам следует знать перед изменением раздела Vista. Главное из них – то, что уменьшение раздела ограничено размещением некоторых непереносимых файлов Windows, таких как файл подкачки или области хранения теневых копий, так что доступное место может быть не так велико, как вы думаете. Есть способ обойти это, но он сложноват и выходит за рамки этого урока. Однако если вы начинаете установку с нуля, то большая часть диска должна быть доступна.

Проверьте место на диске

Диалог усечения покажет объем доступного дискового пространства, возможный размер усечения, пространство для определения нового раздела и окончательный размер раздела **C:**. По умолчанию новый раздел займет все доступное пространство, но мы можем добавить новое значение, используя виджеты в виде стрелок вверх/вниз, или путем ввода числа в строке. Это значение должно выражаться в мегабайтах (МБ), так что 10,240 – это будет примерно 10 ГБ, а 20960 ближе к 20ГБ. В нашем примере мы остановились на 10 ГБ, потому что мы будем иметь дело с действительно объемной информацией – фотографии, видео и музыка – на разделах Windows.

В нашей новой тестовой системе изменение размера заняло около минуты и, к изумлению, не потребовало перезапуска по завершении! В результате получился новый раздел 10,2 ГБ, который Vista определила как неиспользуемое место. Если бы мы работали только в Windows, то сейчас его можно было бы преобразовать в раздел NTFS. Но мы добавляем Linux, так что пришло время выключить Vista и вставить CD с Ubuntu в привод новой машины. Вы можете, естественно, выбрать любой из массы дистрибутивов Linux, так что хорошо бы немного поискать и почитать об опыте других на подобном оборудовании перед тем, как выбрать идеальную для вас ОС.

Обновляем Vista

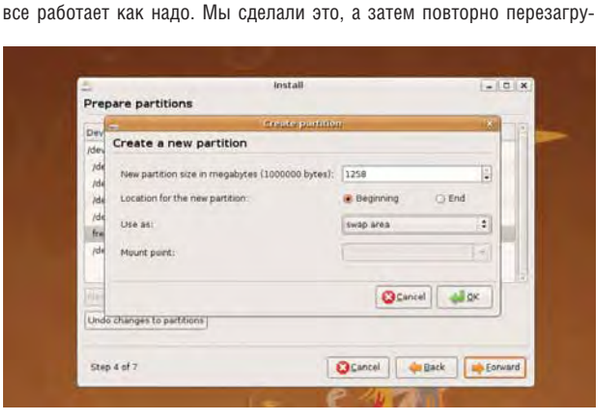
Наша новая машина оснащена удобной опцией вызова меню загрузки клавишей F12, так что для запуска Ubuntu мы нажимаем ее, а затем выбираем DVD-привод из предлагаемого списка. Если это не ваш случай, потребуется обратиться к вашей документации, чтобы выяснить, как поменять порядок загрузки. На некоторых машинах действует удержание клавиши C, а на других может потребоваться войти в настройки BIOS.

После загрузки Live-дистрибутива, перед тем как щелкнуть по кнопке Установить, мы можем проверить, что сеть, звук и видеокарта работают так, как требуется. Здесь пригодятся Live-дистрибутивы (и **LXFDVD**), потому что хорошее первое впечатление от того, как работает дистрибутив, можно получить, даже не дав ему притронуться к вашей системе.

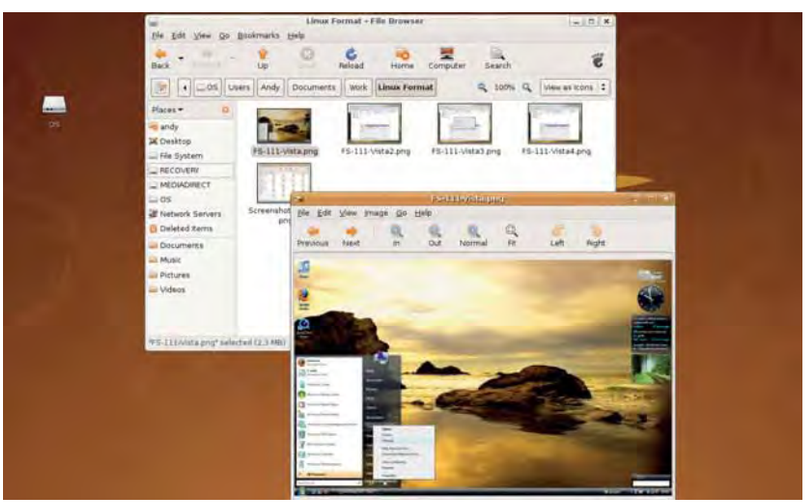
Добавляем swap-раздел

Для запуска апплета установки, в Ubuntu дважды щелкаем на иконке Install, расположенной на рабочем столе, затем, до менеджера разделов, проходим экраны местоположения и настройки клавиатуры. В нем много опций, но мы выберем Manual [Вручную], поскольку раздел нас уже ждет. В менеджере разделов выберите доступный раздел и нажмите New Partition [Новый раздел]. Немного уменьшите размер этого раздела – оставив 512 МБ для подкачки – а затем выберите тип файловой системы (вероятно, лучше будет оставить стандартное назначение) и определите точку монтирования как **/** (он станет корневым разделом). Теперь выберите последний кусок свободного места, дважды щелкните на нем и установите пункт Использовать как в значение Раздел подкачки.

Создав разделы, можно переходить к обычной работе по установке, например, созданию пользователей, а затем, наконец, к копированию файлов. В нашем случае установка заняла около 15 минут (причем мы продолжали работать в Ubuntu), и после перезагрузки мы смогли выбирать между Ubuntu и Vista. После входа в новую систему Linux вы обнаружите в *Nautilus* или *Konqueror* различные разделы Windows Vista. В Ubuntu на нашей машине, к примеру, основной раздел Vista и вся информация на нем были доступны в Places > OS [Места > OS]. Открыв его, мы перемещались по файловой системе, открывали и сохраняли файлы и перемещали элементы как обычно. Несмотря на искушение остаться в Linux, ненадолго вернитесь в Vista, чтобы убедиться, что все работает как надо. Мы сделали это, а затем повторно перезагрузились в Vista после передачи нескольких мегабайт файлов из Linux в файловую систему Windows. Когда вы убедитесь, что все работает, вы можете выполнить более объемные переносы, с уверенностью, что все будет в порядке



Выберите подходящий раздел для размещения установки Linux.



Разделы Windows должны быть доступны в вашем стандартном файловом браузере.

Не бойтесь!

Процесс разбиения жесткого диска для Linux был чреват ужасом и опасностями – особенно страшно было остаться с полностью нерабочей машиной, на которой не грузятся ни Windows, ни Linux. Однако добавление удобной системы разбиения диска в Vista и продолжающиеся усилия разработчиков дистрибутивов в русле предотвращения потери данных в процессе установки принесли действительно реальные дивиденды, и вся работа становится гладкой и безопасной, как шелковые спортивные трусы, укрытые в форте Нокс.

Как отмечалось выше, процесс установки в параллель с Windows XP на этапе разбиения больше зависит от выбранного вами дистрибутива, потому что здесь нет «родного» встроенного в ОС инструмента. Вы, конечно, можете купить для этих целей приложение вроде *Partition Magic*, но в ходе сотен установок на Windows-компьютеры **LXF** пока так и не остался с полностью сломанной машиной. Однако это не означает, что такого не может случиться, поэтому чрезвычайно важно обеспечить наличие актуальной резервной копии всех данных, которые вы бы не хотели потерять. LXF

Скорая помощь

Соберите коллекцию Live-дистрибутивов и все их опробуйте, прежде чем остановиться на какой-либо ОС.

Не сжигайте мосты

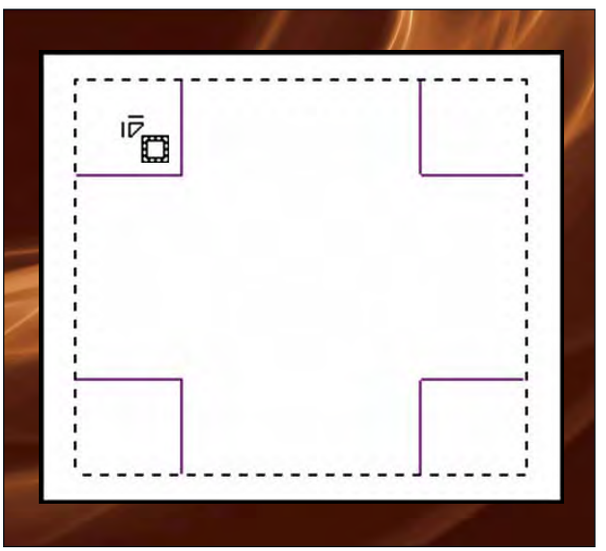
Вы можете, конечно же, полностью очистить вашу новую машину и установить один лишь Linux. Это будет немного проще, но помните, что если вам понадобится доступ к любому Windows-приложению (особенно играм), придется воспользоваться приложениями на базе *Wine*, например, *CrossOver*, или сохранить раздел Windows для эпизодических загрузок.

Через месяц Несколько способов сделать вашу Linux-машину более защищенной.

GIMP Графический редактор с открытым кодом, в который вы тоже можете вонзить зубы

GIMP: выделяй

Кто ж не знает, что такое выделение! Но хорошо ли вы это знаете? На сей раз **Майкл Дж. Хэммел** колдует над несколькими корявыми изображениями.



› **Можете зацепить активные точки для редактирования изображения, пока отображаются границы выделения.**

Наш эксперт

Майкл Дж. Хэммел является участником проекта *GIMP* и автором трех книг по данной теме, включая последнюю *The Artist's Guide to Gimp Effects* [Введение в эффекты *Gimp* для художников].

Если маска – сердце композиции, то выделение – кровь, дающая ему жизнь. Для тех, кто не в курсе, композиция – это просто процесс соединения двух изображений. Без маски она превратится в детский коллаж, который можно склеить, вырезав картинки из журналов или газет. Но маска позволяет не только наложить одно изображение поверх другого. Она, вместе со своими компаньонами – слоями – позволяет размывать границы верхнего и нижнего изображений, превращая их в единое целое.

Есть много способов создавать маски, и почти все они начинаются с хорошего выделения. Умение превращать выделение в полезные маски – это изящное искусство сочетания выбранных частей. Работая с множественными выделениями, мы можем очертить практически любую форму, от квадратов до ворсинок меха.

Я начну этот урок с обзора основ выделения, а затем перейду к филигранной технике более сложных случаев. Я также коснусь методов усовершенствования выделения, чтобы маска или слой могли более ровно переходить в другой слой. Набор основных инструментов выделения был пересмотрен в *GIMP 2.4*, и теперь они снабжены более дружественным интерфейсом. Инструменты Прямоугольник [Rectangle] и Эллипс [Ellipse] окружают объект границей с активными точками для интерактивного редактирования выделения. Перемещайте мышь по границам выделения – влево, вправо, вверх, вниз и к углам – чтобы увидеть активные точки. Заодно обратите внимание на изменения внешнего вида курсора, которые отображают направление, куда можно перетаскивать, чтобы изменить выделение.

Выделение активно, пока его граница находится на своем месте. В этом случае можно, например, перетащить в выделенный объект цвета с панели инструментов, чтобы выполнить ровную заливку. Однако вам может понадобиться завершить редактирование изображения, щелкнув внутри выделенной границы или нажав клавишу Enter. Завершение выделения исключает возможность его правки, поскольку граница исчезает. Редактирование прямоугольного и эллиптического выделения ограничено изменением высоты и ширины при использовании активных точек на границе выделения.

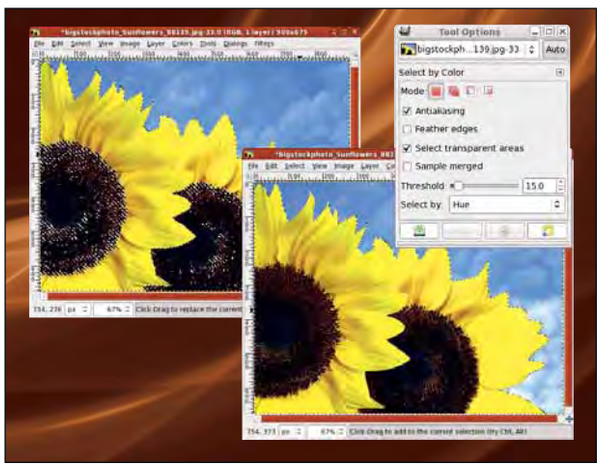
Квадратики в углу

К счастью, последняя версия *GIMP* дает больше возможностей для редактирования. К релизу 2.4 добавлена возможность трансформировать выделение после его завершения. Инструменты преобразования (Rotate [Поворот], Scale [Масштаб], Shear [Искривление], Perspective [Перспектива] и Flip [Отражение]) выполняют трансформацию текущего слоя, активной границы выделения или контура. Это означает, что можно создать выделение с помощью инструмента Свободное выделение [Free Select] или «лассо» и добавить ему перспективы с помощью инструмента Perspective. Но здесь следует быть осторожным. Чтобы инструменты преобразования работали с выделением, их надо настроить, нажав соответствующую кнопку в диалоге Параметров инструментов. Первая расположенная в нем кнопка применяет преобразование к изображению, вторая – к текущему выделению, а третья – к активному контуру. Перед выполнением преобразования убедитесь, что нажата именно та кнопка, которая вам нужна.

В примере справа вверху (подсолнухи) я хочу выделить цветки. Можно использовать Select By Colour [Выделение по цвету], чтобы пометить небо, а затем инвертировать эту область. Но на небе есть бледные облачка, которые не отделить от цветков в нижней правой части изображения. К счастью, можно начать работу и с подсолнухов.

» **Месяц назад** Мы сопрягли слова и изображения в единое целое, как лошадь и экипаж.

и властвуй



› **Сочетание выделения на основе цвета и выделения вручную не требует особой точности.**

Выбрав на панели инструментов инструмент Select By Colour, я выбираю в меню Выбор по: [Select By] пункт Тон: [Hue] и щелкаю на часть подсолнухов желтого цвета (без перетаскивания: просто нажимаю). Это выделит границы лепестков, но некоторые семечки останутся неохваченными. Поскольку линия, обводящая лепестки, является границей выделения, все, что вам нужно – это добавить сюда инструмент Свободное выделение [Free Select].

Оставив текущее выделение без изменений, я выбираю инструмент Free Select. Затем проверяю, чтобы в диалоге Параметры инструментов был активен режим добавления (второй слева пункт в ряду Mode [Режим]) и, наконец, рисую свободное выделение вокруг семечек, при необходимости вылезая мышью за границы холста.

Отделение объекта

Смысл выделения заключается в том, чтобы вычленить из изображения некий объект. Способ, как это сделать, во многом зависит от исходной картинки. Неоднородный фон усложняет процесс, но не делает его невозможным. Хитрость работы с изображениями такого типа – в использовании более мудреных и тонких методов. Инструмент Умное выделение [Fuzzy Select], иногда именуемый «волшебной палочкой», работает во многом похоже на Выделение по цвету, но он даже больше подходит для изображений с пятнами ровного или почти ровного сплошного цвета. Фактически, как бы ни было легко выделить цветки с помощью инструмента Выделение по цвету, Умное выделение сделает это в одно движение.

Выберите инструмент Fuzzy Select на Панели, затем нажмите и медленно тащите по небу на изображении (небо, не цветы!). По мере растягивания выделение все больше разрастается, пока не захватит все небо. Теперь инвертируйте выделение (Select > Invert [Выделение > Инвертировать]), чтобы обвести цветы. Из-за разницы оттенков именно в этом конкретном изображении выделение получается исключительно точным.

Иногда граница между предметом и фоном видна явно. В подобных случаях применяется инструмент Ножницы [Scissors], чтобы обвести объект серией контрольных точек-узлов, как при создании контура. Этот процесс немного более трудоемок, но он позволит выделить более мелкие детали объекта и, что важнее – перемещать точки и редактировать абрис перед тем, как конвертировать контур в выделение.

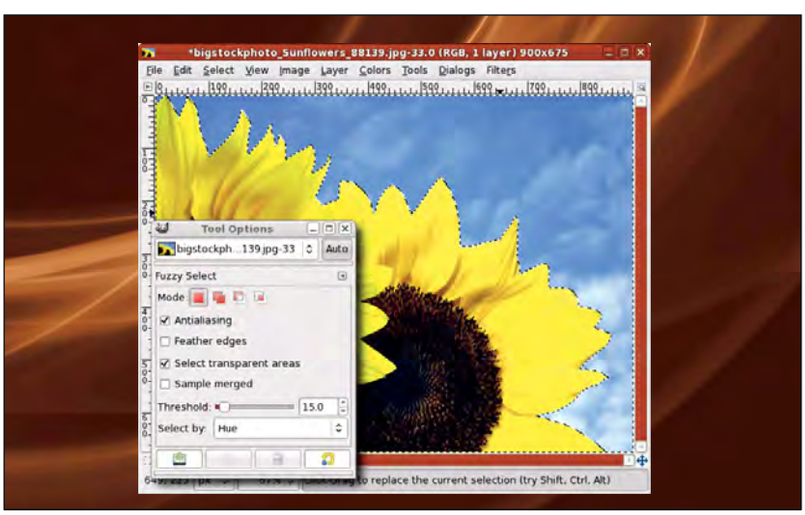
Умные ножницы

Инструмент Scissors [Ножницы] работает, находя границу поблизости от того места, где вы кликнули. На эту границу ставится узел, а между предыдущим и последующим узлом рисуется кривая. Обведя контур предмета, вы нажимаете в любом месте внутри обводки, чтобы конвертировать ее в выделение. В качестве альтернативы можно использовать контуры для создания похожего выделения. Преимущество контура в том, что он обитает в собственном диалоге (подобном диалогу Слои), и потом вы сможете его редактировать, изменяя выделенный объект.

Хотя этого здесь не показано, на холсте контур не присутствует: он существует только в диалоговом окне Контуры. Чтобы он появился на холсте, вам надо или обвести его (тогда он появится в текущем слое) или конвертировать его в выделение. Обе опции отображены в виде кнопок в диалоговом окне Параметров инструмента Контуры.

Самое последнее добавление к коллекции инструментов выделения *GIMP* – это Foreground Select [Выделение переднего плана]. Данный инструмент, так же, как и Ножницы, обладает встроенным интеллектом, но использовать его проще. Будучи выбранным из Панели инструментов, Foreground Select превращает курсор в лассо. Используйте его, чтобы создать приблизительный абрис объекта. На примере со следующей страницы я обвел строение, выйдя за границы холста слева и вновь войдя в границы справа. Очертания имеют форму грубого овала и захватывают часть фона.

По завершении выделения с помощью лассо Foreground Select окрашивает невыделенный фон в голубой цвет. Покончив с работой лассо, курсор принимает вид круглой кисти. Диаметр кисти можно увеличить через Параметры инструментов (этот шаг необходим, если вы работаете с очень большим изображением, поскольку размер кисти по умолчанию рассчитан на web-изображения, а не на печатные). Теперь кликните по объекту за пределами переднего плана и протащите мышь, чтобы нарисовать линию, проходящую через объект на переднем плане. Линия должна охватывать пиксели, дающие представление о цветах объекта на переднем плане. Когда вы отпустите кнопку мыши, инструмент Foreground Select начнет вычислять контуры объекта на переднем плане. Чтобы завершить выделение, нажмите клавишу Enter, »



› **Указав Выбор по Тону [Select by Hue], вы поможете Умному выделению определить различия в оттенках цвета неба и подсолнухов.**

» и вы увидите вокруг границ выделения шеренгу ползущих муравьев.

Полезность инструмента Foreground Select заключается в простоте создания выделения: все, что вам нужно – грубое очерчивание и кисть. Недостаток инструмента – медлительность при работе с кистью, если она толстая. Несмотря на это, Foreground Select способен создавать исключительно точные выделения при незначительных затраченных усилиях.

Альтернативные методы

Сложные случаи выделения почти всегда подразумевают наличие волосков, меха или иных очень тонких линий. Даже самые продвинутые инструменты сами по себе не умеют создать такое выделение. Так что придется комбинировать инструменты или использовать специализированные техники маски, обойдясь вообще без инструментов выделения.

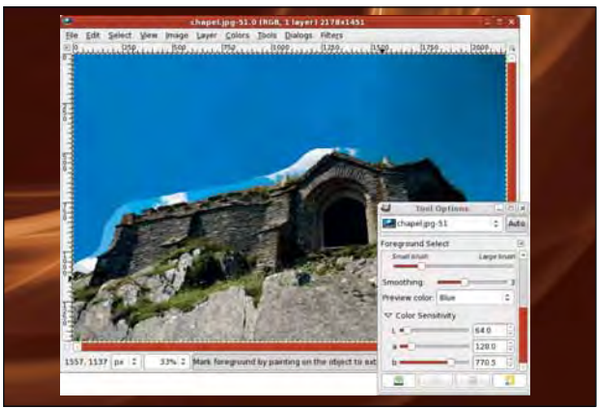
Испробуем Выделение переднего плана на более сложном изображении – наборе теннисных мячей, прикорнувших на сетке теннисной ракетки (внизу справа). Это изображение более сложное, поскольку мячи не соприкасаются и их пиксели нельзя обвести одной непрерывной линией, как это делалось в предыдущем примере. Хуже того, края ракетки тоже желтого цвета. К счастью, изолировать мячи можно несколькими инструментами. Сперва я обрисую их контуры с использованием лассо Foreground Select, прилагая все усилия, чтобы включить как можно меньше от ракетки. Затем нарисую линию выделения объекта на переднем плане через все теннисные мячи. Это отлично работает для выделения всех мячей – и только мячей; но заодно захватывается и часть сетки на ракетке, а вот штамп на ближайшем мяче сюда не попадет.

Теперь я выключаю кнопку Contiguous [Смежная область] в Параметрах инструментов. Затем рисую еще несколько линий для выделения объектов на переднем плане в разных частях мячей – например, там, где темный штамп. Это немного усовершенствует результат Выделения переднего плана, но работы осталось еще много.

Переключаюсь на Select Background [Выделение фона] и прорисовываю очертания мячей поверх сетки ракетки. Сначала я увеличил изображение, чтобы лучше его видеть. Размеры кисти при увеличении изображения не меняются, и для более тонкой и детализированной работы лучше подойдет маленькая кисть. Я уменьшаю изображение и нажимаю Enter, чтобы конвертировать это все в выделение. И тут я вижу, каким убогим оно получилось. Придется подчистить его с помощью Быстрой маски [Quick Mask].

Действие Быстрой маски очень похоже на результат Выделения переднего плана. Когда вы ее активируете (кнопка Quick Mask – это квадратная кнопка внизу слева окна изображения), все, что находится за пределами выделения, окрашивается в красный цвет. Красный – выбор по умолчанию: вы можете щелкнуть правой кнопкой мыши по кнопке Quick Mask, чтобы указать другой цвет, однако красный отлично сработал с желтыми теннисными мячами.

› Когда вы используете инструмент Ножницы, кривая между узлами выходит не особо гладкой, так что с этим выделением лучше использовать контур.



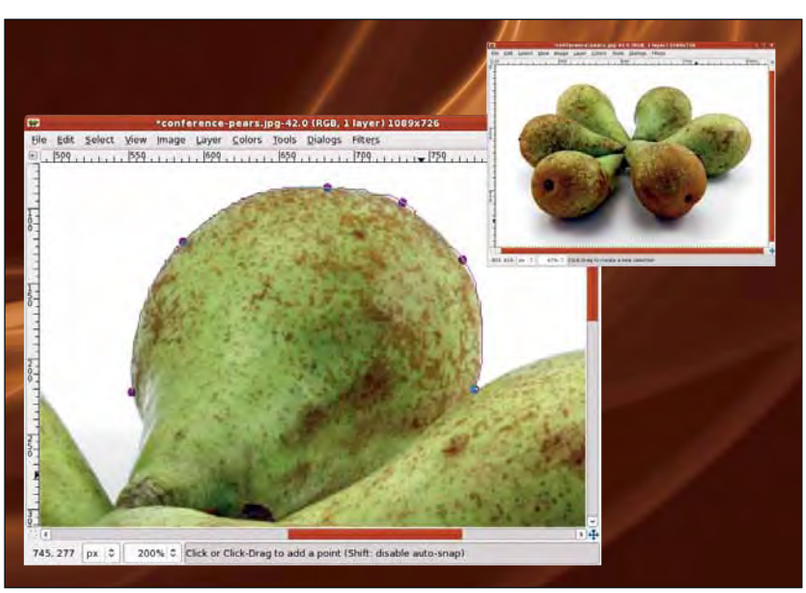
› Лассо просто создает первоначальное выделение, которое потом будет усовершенствовано с помощью Выделения переднего плана.

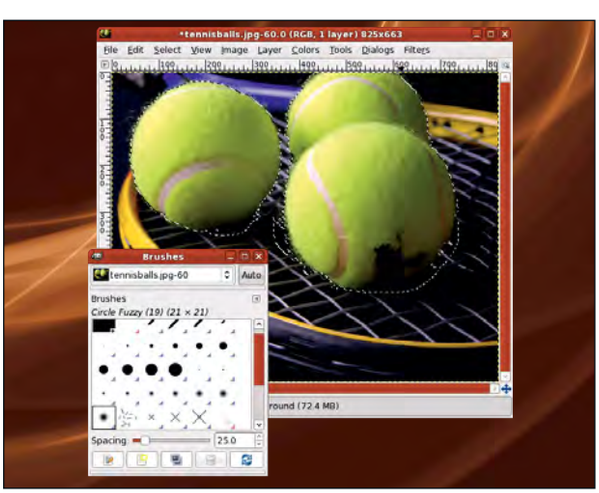
Далее вы выбираете из Панели инструмент работы с цветом. Идеально подходит Paintbrush [Кисть] в сочетании с мягкими краями – например, одной из круглых размытых кистей Circle Fuzzy, входящих в *GIMP* по умолчанию. Кисти с мягкими краями в Quick Mask действуют как растушевка контура выделения; а если вы решите использовать кисть с жестким краем, растушевать получившееся выделение можно будет и после (используя Select > Feather [Выделение > Растушевать]).

Теперь мы просто закрасим мячи белым, чтобы добавить их к выделению. При использовании Quick Mask, белый цвет снимает маску (он на самом деле не виден), а черный добавляется к маске (на месте покраски появляется оттенок маски). Я закрасил примерные контуры мячей белым, а затем переключился на черный, чтобы удалить области, которые не хотел включать в выделение. С помощью Quick Mask становится возможным увеличить изображение, использовать очень маленькую кисть и вручную выделить тонкий мех или волосы на изображении. Это – очень трудоемкий процесс, и, наверное, в нем нет необходимости, если не заниматься созданием изображении с высокой степенью точности; просто приятно знать, что вы такое можете. Закончив заполнение Быстрой маски, я снова нажимаю на кнопку Quick Mask, чтобы переключиться на настоящее выделение.

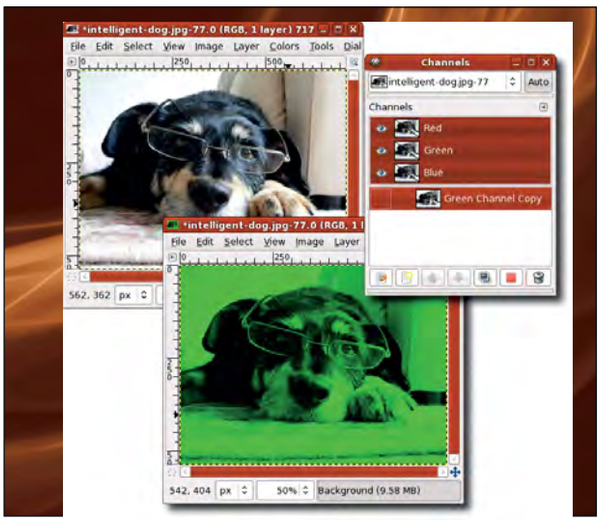
Исключительные случаи

И, наконец, рассмотрим те изображения, с которыми просто нельзя работать при помощи основных инструментов выделения: в таких случаях приходится все делать вручную. Возьмем, например, этого





› Быстрая маска позволяет закрасить области, которые будут выделены (или не выделены). Кисть с мягкими краями растушует контур выделения.



› Для данного изображения выбран зеленый канал, хотя красный в этом случае тоже сработал бы.

ученого пса на верхнем изображении. Несколько пассов с помощью Выделения переднего плана показывают, что данный инструмент просто не работает без солидного редактирования. Из-за времени, требуемого на обсчет каждого нового движения мышью для добавления очередного объекта в выделение на переднем плане, это прочесывание становится чересчур затратным по времени, да еще и с необходимостью постоянной корректировки.

Более простое решение – найти канал, обеспечивающий наибольший контраст, и использовать его как отправную точку для создания маски. Откройте диалоговое окно каналов и отключите их все, нажав на значок видимости. Затем снова включите их, чтобы найти тот, в котором самый сильный контраст. Для данного изображения это будет зеленый канал. Кликните по нему каналу правой кнопкой мыши и выберите Duplicate Channel [Создать копию канала]. Копия будет добавлена в конец списка. Нажмите на значок видимости (кружок с глазом слева от миниатюры), чтобы включить отображение копии канала.

Используем эту копию канала для создания слоя маски, чтобы изолировать нашего друга-собакина. В окне слоев отключите отображение фонового (и единственного) слоя. Теперь вы можете видеть сам канал, который выглядит, как обесцвеченная версия слоя с собакой. Убедитесь, что канал активен, щелкнув по нему в окне Channels [Каналы], затем откройте окно Levels [Уровни]. Двигайте бегунки, пока не добьетесь наилучших очертаний изображения собаки – забудьте про внутреннее содержание этого изображения, его мы добавим позднее.

Нам ни к чему лишняя грязь вокруг контуров, так что мы можем покрасить их белым прямо в канале. Но пока что у нас нет четкого контура собаки, чтобы заполнить его черной заливкой. Исправить это можно следующим образом: скопировать канал (Select > All [Выделить > Все], Edit > Copy [Правка > Копировать]), вернуться в окно Слои и нажать на фоновый слой, сделав его активным, затем вставить копию в окно изображения (Edit > Paste [Правка > Вставить]). Закрепите это выделение в окне Слои (Layer > New [Слой > Создать]).

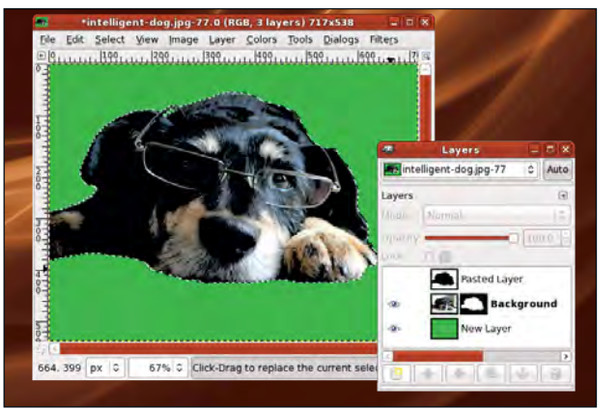
Теперь можно отредактировать этот новый слой, уменьшив его непрозрачность (переместите бегунок Opacity на **25.0** в окне Слои) и аккуратно проведя черную линию по контуру пса там, где на новом слое не отобразилось черного. Здесь я использовал увеличение (чтобы более четко видеть контур), очень маленькую кисть с жестким краем и инструмент Pencil [Карандаш]. Замкнув контур, мы сможем выделить большую часть пространства, окрашенного белым, с помощью инструмента Умное выделение, увеличить выделение на несколько пикселей и заполнить его черным. Затем подчистим остальную часть изображения вручную, с помощью кисти.

Заполненный черным собачий силуэт в новом слое мы можем скопировать в маску оригинального слоя. Щелкните на слое с изображением собаки, сделав его активным, затем добавьте к нему слой маски (Layer > Mask > Add Layer Mask [Слой > Маска > Добавить слой-маску]). Щелкните на новом слое с черным силуэтом собаки в окне Слои, чтобы он стал активным. Скопируйте его (Правка > Копировать), затем щелкните по слой-маске с изображением собаки – убедитесь, что вы щелкнули по самой маске в окне Слои! Вставьте скопированное в маску (Layer > Paste [Слой > Вставить]) и закрепите (Layer > Anchor [Слой > Прикрепить]). Черный в маске замещает прозрачный, поэтому инвертируйте цвета (Colours > Invert [Цвета > Инвертировать]) в маске. В этом пункте к маске стоит также применить легкое размытие (Filters > Blur > Gaussian Blur [Фильтры > Размытие > Гауссово], примерно пять пикселей). Теперь у вас должен получиться хороший контур собаки. Я добавил зеленый фон, чтобы можно было увидеть результаты. Чтобы создать выделение отсюда, просто конвертируйте маску в выделение (Layer > Mask > Mask to Selection [Слой > Маска > Маска в Выделение]).

Все дело в выборе

Выбор инструментов и техники существенно зависит от исходного изображения. Изображения с высокой контрастностью отлично обрабатываются с помощью основных инструментов. Большая разница в цвете фона и объекта на переднем плане хорошо поддается обработке с помощью Fuzzy Select, Select By Colour и Foreground Select. Объекты, сливающиеся с фоном и имеющие множество тонких линий вдоль контура, и/или расположенные перед загроможденным пестрым фоном, требуют более тщательной работы с использованием разных инструментов, настроек и уточнений в Уровнях [Levels] и настроек Яркости/Контраста [Brightness/Contrast] для создания подходящей маски.

Короче, лучше всего действует метод проб и ошибок. Будь на свете универсальный магический инструмент, уж конечно он был бы в *GIMP*. К счастью, ящик с инструментами *GIMP* полон разнообразными орудиями, позволяющими легко и непринужденно экспериментировать с разными способами выделения. DXF



› Создание подобного выделения требует гораздо больше усилий, но результат того стоит.

Настройка кисти

Основные кисти в *GIMP* довольно стандартны по размеру. Вы можете создавать их дубликаты, чтобы редактировать форму и размер, но мне показалось проще всего создать новую кисть, которую я редактирую вручную, каждый раз, для того процесса, над которым я работаю. Создать новую кисть можно с помощью кнопки New Brush [Создать кисть] в панели Brushes [Кисти].

» **Через месяц** Шаг в сторону: научитесь устранять дефекты на своих цифровых снимках.

Мультимедиа: HDR

Марко Фиоретти начинает серию уроков по мультимедиа, с советами и новостями от *DigiKam*, музыкального сервера *Icecast*, *LMMS* и забавным «пожирателем времени» от *Blender*.

Linux Format регулярно освещает темы домашнего мультимедиа, от архивирования ваших цифровых фотографий или музыкальных коллекций до производства и проигрывания семейных видеороликов или онлайн-подкастов. Однако информация такого рода устаревает с выходом новых версий программ – а ведь Linux был бы не Linux, если бы они не появлялись каждые пару недель. По той же причине, например, представляется невозможным досконально освещение всех способов использования аудиоприложений в одном учебнике. На самом деле, учебник украсился бы массой подробностей, если бы только они успевали попасть в приложение на том этапе, когда мы о нем писали.

Я собираюсь восполнить несколько пробелов по части ряда интересных функций, которые мы проскочили из-за огромных темпов развития свободного ПО, не повторяя объяснения про старые, уже описанные нами возможности – вы о них и так знаете. Поле для деятельности здесь огромное, и если вы хотели бы познакомиться с какими-то моментами более подробно, напишите нам (именно так появилась на свет статья о *MythTV* на стр. 32). Итак, без дальнейших прелюдий займемся…

Наш эксперт

Марко Фиоретти
Автор *The Family Guide to Digital Freedom* [Семейное руководство к цифровой свободе], активный сторонник и разработчик свободного ПО.

PFSTools в HDR-фотографии

В LXF105 Дейл Стрикленд-Кларк представил замечательный мир изображений с расширенным динамическим диапазоном (HDR). Данная технология позволяет объединять разные фотографии одного и того же объекта для создания новой версии с большим количеством деталей в тенях и световых участках, чем в любом из составляющих снимков. HDR-фотографии обычно сохраняют в формате OpenEXR (**www.openexr.com**), позволяющем держать несколько экспозиций одного кадра в одном файле.

Обычная в HDR процедура – это наложение тонов (tone mapping), методика, позволющая снижать контрастность до диапазона, отображаемого на экране или бумаге, при сохранении детализации и цветовой гаммы. В последнем релизе обработчика HDR, *PFSTools* (**http://pfstools.sourceforge.net**), включен новый инструмент для этой операции. Посмотрим, как он работает.

Стандартная процедура выглядит так, как изображено во врезке внизу: на рисунке справа детали двух экспозиций объединены. Магическое заклинание для командной строки, чтобы провернуть такой же трюк с вашими фотографиями в *PFSTools*, выглядит следующим образом:

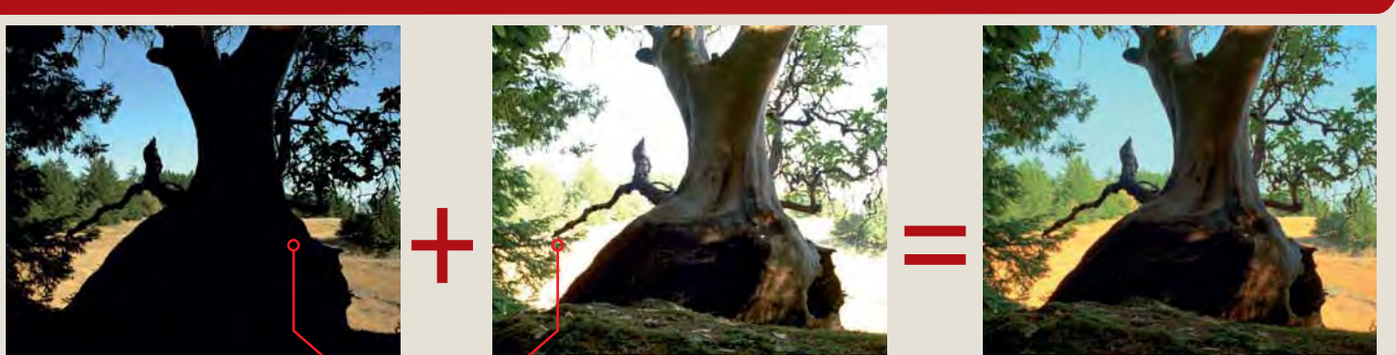
```
pfsin tree.exr | pfstmo_mantiuk08 -v -e 1.5 -c 0.7 | pfsview
```

что в переводе на человеческий язык означает:

» Распаковать все экспозиции фотографии из файла OpenEXR (**tree.exr**) с помощью *pfsin*.

» Отправить их в несжатом формате в программу *pfstmo_mantiuk08*,

Наложение тонов с HDR



Дерево с низкой выдержкой

Дерево с высокой выдержкой

Дерево с наложением тонов, когда детали обеих фотографий собраны воедино.

Оба снимка хранятся в файле **tree.exr**.

и DigiKam

Наложение тонов при рассеянном свете

Исходное детальное HDR-изображение со стандартным наложением тонов…

… и то же изображение в рассеянном свете на экране ноутбука.

HDR-наложение тонов сделано с учетом рассеянного света.

которая и выполняет наложение.

» Отобразить результат во всей красе в просмотрщике изображений *pfsview* (вы также можете использовать его для редактирования и сохранения настроек экспозиции).

Адаптивное наложение тонов еще круче. Взгляните на вторую врезку, вверху. Слева – обычный снимок в формате JPEG, как он выглядел бы на вашем мониторе при оптимальных условиях, без каких-либо отражений и других искажений от рассеянного света. Если вы примените к нему стандартную процедуру, описанную выше, и будете просматривать его на экране ноутбука в полдень на вашей веранде, результат будет похож на второе изображение, которое выглядит блеклым.

Чтобы исправить эту беду, нужно выполнить наложение тонов, адаптирующееся к настройкам вашего монитора; при этом картинка преобразовывается таким образом, чтобы компенсировать недостатки монитора и свести на нет эффекты, связанные с рассеянным светом (эту информацию приложение получает в реальном времени, если на мониторе есть сенсор рассеянного света). В результате изображение выглядит как HDR даже при полном свете дня. Команда для получения этого эффекта выглядит следующим образом:

```
pfsin japan_garden.exr | pfstmo_mantiuk08 -d a=5000 |
pfsdisplayfunction -I -d a=5000 | pfsview
```

Она схожа с первой: это совмещение программы наложения тонов с утилитой *pfsdisplayfunction* для получения адаптивного результата.

Для получения более подробной информации об адаптивном к вашему монитору наложении тонов, пройдите по ссылке **www.mpi-inf.mpg.de/resources/hdr/datmo** или напишите одному из разработчиков, Рафалу Мантьюку (Rafal Mantiuk; он любезно поделился с нами этими примерами), на **mantiuk@cs.ubc.ca**.

DigiKam Интеллектуальный поиск

Кстати о цифровой фотографии: разработчики менеджера фотографий *DigiKam* (**www.digikam.org**) переводят эту программу и все связанные с ней библиотеки на KDE 4. Они также предоставляют много новых функций, начиная от интерфейса базы данных до множественных корневых альбомов, прозрачной работы по сети, геолокации и поддержки стандарта Digital Negative (**www.adobe.com/products/dng**), полностью документированного расширения TIFF от Adobe для целиком переносимых необработанных [raw] цифровых фотографий высокого качества. Бета-релизы *DigiKam* для KDE 4 появились в июле 2008 года. Пока им, наверно, рановато вверять свои драгоценные хранилища фотографий, но функции так заманчивы, что не грех и побаловаться с новой *DigiKam* (на копии ваших архивов, естественно) и, возможно, внести свою лепту в отчеты об ошибках.

Полезнейшей из функций является поиск: есть новый интерфейс для нахождения дубликатов и, кроме того, функция Fuzzy Search [интеллектуальный поиск]. Конечно, поиск дубликатов в *DigiKam* уже есть, но в новой версии эта функция должна быть ощутимо быстрее, потому что поисковый модуль теперь встроен в основной код и стал более приятным в использовании. Взгляните на экранные снимки, сравнивающие поиск дубликатов в KDE 3 (вверху) и в KDE 4 (внизу). Довольно гадать над именами файлов: просто нажмите на кнопку Поиск Дубликатов [Find Duplicates], и вы получите слева миниатюры изображений, имеющих дубликаты. Выберите одну из них мышью, и галерея всех дубликатов появится в остальном пространстве окна.

Вейвлеты Хаара

На минуточку побудем умниками: в KDE 4 *DigiKam* хранит во внутренней базе данных «наброски» каждого изображения, сгенерированные с помощью вейвлетов Хаара [Haar wavelets]. Ну как не использовать штуку с таким классным названием? Вообще-то мы упомянули вейвлеты Хаара неспроста: именно они лежат в основе нового инструмента Fuzzy Search. Это метод поиска картинок, которые просто похожи (но не являются дубликатами), либо по выбранной мышью миниатюре, либо – вот это да! – по эскизу, нарисованному от руки! Честно!

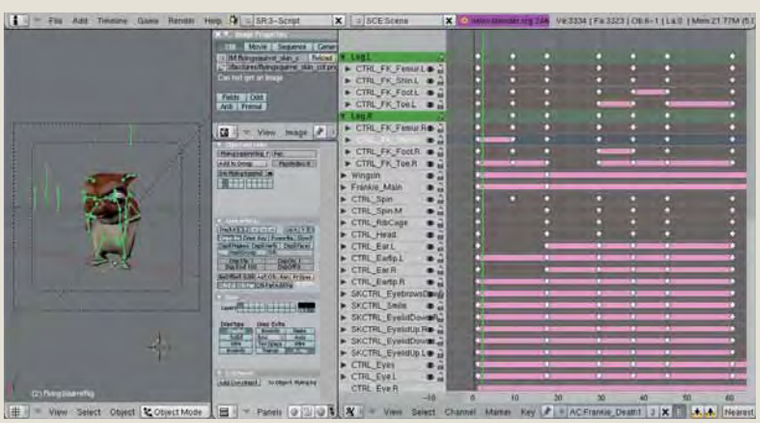
› Галереи эскизов изображений пришли на смену текстовым спискам в функции Поиска дубликатов.

Blender

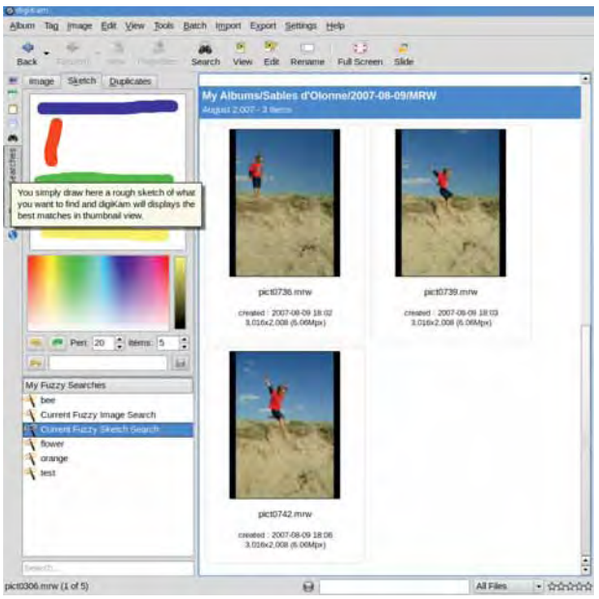
Хотите разработать первоклассную свободную 3D-игру с онлайн-сообществом? ОК, сначала идем на http://apricot.blender.org. На этом сайте, после открытых фильмов *Orange* [Апельсин] и *Peach* [Персик] (www.bigbuckbunny.org), Blender Institute запустил проект разработки открытой 3D-игры с участием главных героев из *Peach*. С главной страницы *Apricot* [Абрикос] можно скачать демо-версию, чтобы с ней поиграть. Это сделано не ради вашего удовольствия, а потому, что разработчики хотят протестировать его на как можно большем количестве графических карт и компьютеров! Когда вы наиграетесь, вам понадобится всего минута, чтобы понять, как делать анимацию с *Blender*, и, возможно, внести свой вклад в развитие игры!

Скачайте файл http://apricot.blender.org/wp-content/uploads/2008/06/frankanim.blend и откройте его в *Blender*: вы увидите окно, похожее на показанное на экранном снимке. Теперь щелкните где-нибудь слева и нажимайте стрелки вправо-влево: нет, вы не перебрали пивка, Фрэнки [Frankie] и вправду двигается.

Это вы только что открыли готовый к запуску файл анимации *Blender* – самый доступный способ научиться анимации, не предваряя открытие *Blender* перелопачиванием тонны руководств. Под текстурой Фрэнки находится скелет из нескольких «костей», которые видны в *Blender* как зеленые линии. Можно отдельно контролировать движение каждой кости, привязав к ней комбинацию клавиш, а не то *Blender* сам назначит клавиши, когда вы двигаете очередную кость. Каждая кость имеет отдельный канал в окне Редактирования действий (на экранном снимке – крайнее справа), которая используется для сложения и контроля, что, где и как двигается. Естественно, можно группировать части тела, чтобы они двигались вместе, связывать простые движения для создания более сложных или зацикливать движения, например, для создания пробежки из отдельных шагов. Пошаговые инструкции, как анимировать модели в *Blender*, находятся по адресу http://wiki.blender.org/index.php/BSoD/Introduction_to_Character_Animation.



› Вдохнуть жизнь во Фрэнки, одного из главных героев *Apricot*, очень легко!



› Новой KDE 4 версии *DigiKam* достаточно ваших смутных воспоминаний о снимке для его обнаружения!

Помните день, когда ваш сынишка играл на пляже? Вы сделали чудесные фотографии, но где они? У вас в памяти сохранились только голубизна неба и красная футболка – обновка, которой он так гордился. Не волнуйтесь, этого достаточно для *DigiKam*: нажмите на Fuzzy Search, нарисуйте голубую и красную полосы (это представление неба и футболки) в панели набросков – и получите искомую фотографию! Знаем, знаем: сама по себе, эта возможность отнюдь не нова, и основной алгоритм (быстрый запрос изображений с разным разрешением) и исходный код взяты из другого менеджера изображений под Linux, под названием *ImgSeek* (www.imgseek.net); он делает то же самое. С другой стороны, конкуренция и конструктивный плагиат в приложениях с открытым кодом приветствуются, а интеллектуальный поиск, интегрированный в среду рабочего стола KDE – это же здорово, верно?

Запомните: вы можете найти эти и другие экранные снимки *DigiKam* в хорошем разрешении на www.flickr.com/groups/digikam-labs, любезно предоставленные главным разработчиком Жилем Колье [Gilles Caulier].

Icecast Транслируйте музыку через Интернет

В LXF106 мы узнали, как создать интернет-радиостанцию в *Icecast*. Хорошая новость: почти сразу после того, как мы написали учебник, и после двух с половиной лет разработки была выпущена новая версия *Icecast* (ее можно скачать с www.icecast.org как исполняемый файл Windows, исходный код или RPM-пакет) Это сервер вещания под Linux и Windows, созданный для трансляции аудиофайлов MP3 или Ogg Vorbis для всех типов клиентов. Полный список новых функций версии 2.3.2 доступен на домашней странице. Две наиболее интересные из них, по крайней мере, для начинающего ди-джея, который хотел бы донести свою музыку до слушателей по всему миру, имея при этом самый полный контроль над трансляцией – это улучшенная поддержка разных алфавитов и возможность создания черных и белых списков IP-адресов.

На практике первая особенность означает, что *Icecast* теперь умеет преобразовывать все метаданные в UTF-8 для мгновенной передачи их в клиенты: в итоге вы можете общаться с большим количеством слушателей на их родном языке. Например, если вы транслируете web-страницы и списки каталогов, посвященные балладам об Афганистане, и некоторые из ваших слушателей используют русскоязычные браузеры, вся передаваемая информация будет отображаться корректно.

Безопасность – это просто

Списки IP-адресов, кроме того, дают вам гибкость другого рода. Задайте в *Icecast 2.3.2* имя списка IP-адресов в текстовом формате, и, в соответствии с конфигурацией, программа будет сама разрешать или запрещать запросы соединений от указанных хостов, не задавая лишних вопросов. Такая функция очень пригодится тем web-ди-джеям, которые непривычны к настройке брандмауэра или не имеют достаточных административных прав, чтобы произвести ее.

DVD-слайд-шоу Создайте видео из ваших фотографий

Если вы не покушаетесь на создание очередного оскароносного «Из Африки», состряпать DVD-слайд-шоу можно довольно быстро. Инструменты *DVD-slideshow* (**http://dvd-slideshow.sourceforge.net/wiki**) – это набор утилит командной строки, которые сделают мультимедийные слайд-шоу на DVD из любой коллекции изображений в формате JPEG или PNG, а также MP3-, OGG- или WAV-аудио-файлов. Накопив нужные изображения и аудиоклипы, всего лишь изучите парочку man-страниц и перечислите последовательность действий в простом текстовом файле. Компоненты *DVD-slideshow* прочитают эту информацию, и ваш шедевр вскорости будет готов.

Последовательность действий описывается рядом строк. Каждая из них ссылается на один видеоэффект или на один аудио- или графический файл и определяет, что с ним делать. Общий синтаксис выглядит следующим образом:

```
[картинка.jpg|ключевое слово]:продолжительность:субтитры:эффект:параметры эффекта
```

где после специального ключевого слова, обозначающего эффект, или имени графического файла вы указываете, используя двоеточие как разделитель, длительность пребывания на экране, требуемую подпись, и далее – несколько дополнительных параметров конфигурации. Не волнуйтесь, вы не обречены на чередование статических картинок: кроме добавления аудиофайла, также можно включить эффекты затухания, увеличения, масштабирования или горизонтального кадрирования. Все возможности детально описаны на man-страницах. Для понимания работы программы взгляните на пример подобных инструкций: это творчески примененная смесь более полных примеров, доступных на домашней странице.

```
# Комментарии оформляются так же, как в скриптах
title:10:Это название слайд-шоу, оно маячит 10 секунд
моя_музыка.mp3:1:crossfade
голос_диктора.mp3:2
родительская_свадьба.jpg:5:День, когда все началось
нервный_жених.jpg:5:По слухам, папа немного нервничал...
все_отплясывают.jpg:5:...но потом всем было ужасно весело
```

Прежде всего отметьте, что поля ‘effect’ присутствуют, только если они нужны. Следующие две строки добавляют фоновую музыку на аудиотреке 1, постепенно затухающую с переходом на голос диктора с трека 2. Воспроизведение пойдет с запуском следующей картинки. Для добавления меню существует другой компонент *DVD-slideshow*, под названием – нипочем не угадаете! – DVD-menu.

В процессе изучения *DVD-slideshow* не забывайте задавать командный ключ -L. Тогда видео будет более низкого качества, со вдвое меньшим разрешением и с более грубыми переходами. Почему нужна подобная гадость? Да хотя бы потому, что ее создание займет на 75% меньше времени. -L-режим прекрасно подходит для изучения и отладки.

LMMS Теперь с Qt4

The Linux Multimedia Studio (*LMMS*, **http://lmms.sourceforge.net**) – это система генерирования звука, реализованная на дизайнерских решениях по типу GarageBand от Apple или FL Studio. Вы сможете составлять сложные музыкальные произведения в два этапа. Сначала сгенерируйте несколько основных звуков с помощью генераторов, включенных в *LMMS*. Подготовив звуки, приступайте к их объединению. Альфа-релизы серии 0.4 *LMMS* уже доступны и выглядят неплохо, почему бы их не попробовать? Посмотрите на снимок с монитора разработчика Тобиаса Дерффеля [Tobias Doerffel], чтобы понять, чего ожидать.

Новый интерфейс пользователя построен на библиотеках *Qt 4*: если вам очень-очень нужно, можете даже запустить *LMMS* на 32-битных версиях Windows. Но хакер, сидящий внутри вас, будет, наоборот, счастлив узнать, что начиная с 0.4.0 версии, *LMMS* можно выполнять посредством скрипта даже без графического интерфейса и даже разделять один *LMMS*-сервер между несколькими пользователями. Кроме того, можно сэкономить массу времени, генерируя и обрабатывая звуки посредством скрипта автоматически. Вот пример:

```
lmms --render foo.mmpz --format wav --samplerate 96000 --oversampling 2 -o foo.wav
```

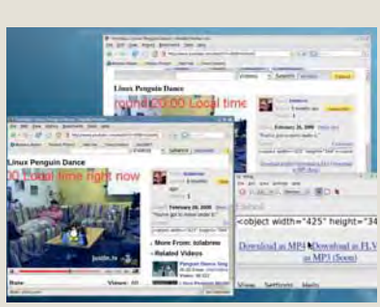
С помощью этого кода вы создадите WAV-версию с частотой дискретизации 96 кГц, обработанную при 192 кГц (частота дискретизации, помноженная на коэффициент запаса точности – oversampling) из необработанного MMPZ-файла. LXF

› Новое поколение *LMMS* включает множество функций по обработке звука, и даже работает под Windows.

Greasemonkey

Greasemonkey (**https://addons.mozilla.org/en-US/firefox/addon/748**) – сверхпопулярное дополнение для *Firefox*, способное запускать сотни разных скриптов для изменения «на лету» страниц, которые вы посещаете. Миллионы людей используют *Greasemonkey* для объединения частей разных web-страниц.

Пользовательский сценарий – это не что иное, как небольшая программа на JavaScript, написанная специально для работы с *Greasemonkey*. Главный репозиторий находится по адресу **http://userscripts.org**. Одно из наиболее полезных недавних дополнений, по крайней мере, для любителей видео, обожающих коллекционировать и совмещать максимум информации – мастер закачек YouTube **http://userscripts.org/scripts/show/30318**. Экранный снимок показывает, как он работает. Слева скрипт в *Firefox* не запущен, справа – запущен. Данный скрипт добавляет, прямо под фрагментом кода ‘Embed this video’, ссылки, просматриваемые в окне *KMag* в правом нижнем углу: Download as MP4 и Download as FLV. Просто нажмите на один из них, чтобы по-быстрому нахватать клипов, готовых для сохранения или редактирования.



› Получение MP4- или FLV-версии видеоролика YouTube теперь ускорилось.

» **Через месяц** Подружите Amarok с Facebook и разберитесь с JPEG’ами.

Календарь: в такт

Как много нужно сделать, как много можно забыть… **Саймон Пикстон** повышает свою организованность, синхронизируя календарь рабочего стола с мобильником.

Если вы тип вроде меня, вам так же сложно запомнить, что запланировано на завтра, не говоря уж о годовщинах свадьбы, днях рождения и даже еженедельных планерках. Вдобавок большинство из нас работают на нескольких компьютерах и разнообразных операционных системах, и чтобы быть организованными, приходится прилагать немалые усилия. Все мои телефоны за последние пять лет имели некое подобие встроенного календаря, но эта функция неизменно была дурно воплощена, требовала использования проприетарного ПО для синхронизации с ПК или признавала только *Microsoft Outlook*.

К счастью, ситуация улучшается, но все равно далека от совершенства. Если телефон продается с программным обеспечением, оно, как правило, создано для работы только с Windows, а клиент календаря по умолчанию – нечестивый *Outlook*. Правда, поскольку множество телефонных календарей совместимы со стандартом iCalendar, обычно можно задействовать и другие календарные приложения. Вдобавок большинство телефонов имеют возможность синхронизации через Bluetooth, и вам не потребуются USB-кабели или драйверы. Некоторые телефоны могут даже синхронизироваться через беспроводное интернет-соединение. На данном уроке мы рассмотрим синхронизацию календаря рабочего стола не только с телефоном, но и с Google Calendar и календарями на других компьютерах, независимо от используемой ОС.

Наш эксперт

Саймон Пикстон
До того влюблен в свой мобильный телефон, что использует его дома для контроля выключателей и розеток.

Часть 1 Mozilla Sunbird

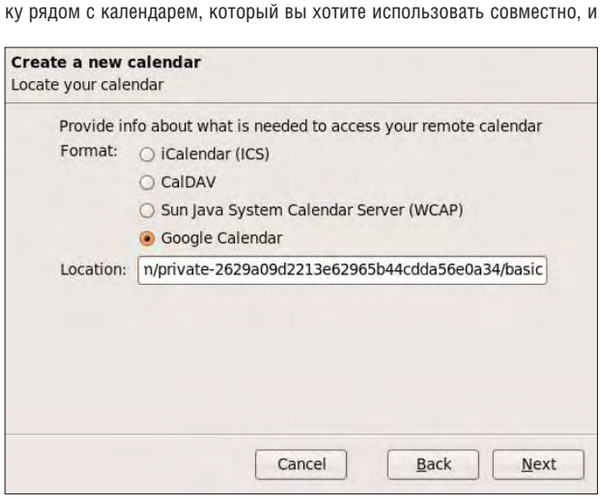
В Ubuntu возможен выбор из двух календарей: *Evolution* и *Mozilla Sunbird/Thunderbird*. Наша цель – создать простой способ синхронизации календаря под Ubuntu, OS X и Windows, а способ ее осуществить – использовать Google Calendar в качестве посредника. Мы начнем с *Mozilla Sunbird*, поскольку это самое простое решение, затем перейдем к *Evolution*.

Sunbird – самостоятельное календарное приложение от Mozilla, его можно скачать с сайта www.mozilla.org/projects/calendar/sunbird. Как и все приложения от Mozilla, оно кросс-платформенное и предлагает одинаковый интерфейс для всех используемых ОС. Кроме того, вместо запуска трех отдельных приложений – *Firefox*, *Thunderbird* и *Sunbird* – можно встроить функционал *Sunbird* в *Thunderbird* с помощью дополнения *Lightning* (www.mozilla.org/projects/calendar/lightning), приведя его в большее соответствие с *Evolution*.

Объединять, а не разделять

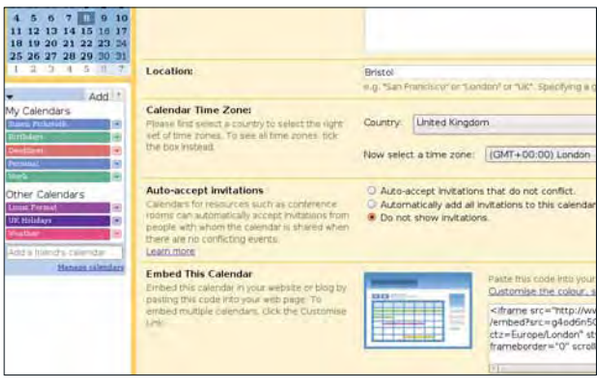
Для установки *Lightning* скачайте XPI-файл, запустите *Thunderbird*, перейдите в Инструменты > Дополнения > Установка [Tools > Addons > Install] и укажите место, где вы сохранили модуль. После перезапуска *Thunderbird* календарь будет встроен, и вы сможете переключаться между ним, почтой и задачами, используя три иконки в нижнем левом углу в *Thunderbird*. Иногда это приводит к проблемам с отображением, поэтому на данном уроке мы используем чистый *Sunbird*.

В *Sunbird* есть два способа использования Google Calendar: подписка или синхронизированный календарь. Подписные календари доступны только для чтения – это, несомненно, полезно для встраивания чужих календарей в свой, но для нашего проекта подходит меньше. Если вам как раз и нужен режим только для чтения, потребуется всего лишь адрес страницы ICS-файла календаря, на который вы хотели бы подписаться. Чтобы получить его в Google Calendar, нажмите на стрелку рядом с календарем, который вы хотите использовать совместно, и



Для добавления Google Calendar воспользуйтесь Provider, расширением *Sunbird*: просто укажите адрес XML-данных календаря.

с телефоном



› Общий доступ к данным Google Calendar можно сделать в Настройках, используя кнопки внизу экрана.

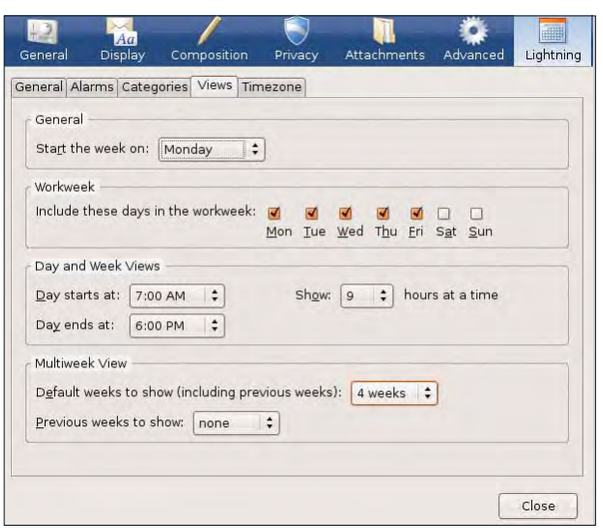
Простота — в совмещении

Хотя для первой части нашего урока мы использовали *Sunbird*, вы, разумеется, можете встроить функции *Sunbird* в *Thunderbird*, чтобы не запускать календарь и почтовый клиент по отдельности. У нас была проблема с модулем *Lightning* при интеграции в *Thunderbird* в первый раз, но после удаления и переустановки модуля все заработало корректно. Для изменения настроек календаря нажмите на Правка > Настройки [Edit > Preferences] и затем на иконку *Lightning*. Хотя вы можете использовать функцию категорий для классификации разных событий, все, что вы получите – это цветную линию вокруг события. Причем линия настолько тонка, что в зависимости от цвета может быть вообще не видна. Мы пришли к выводу, что намного проще создать по календарю на категорию и дать им соответствующие имена – например: Дни Рождения, Персональный или Работа.

выберите Установки календаря [Calendar Settings]. Внизу страницы, в Адресе календаря [Calendar Address], нажмите правой клавишей мыши на зеленую кнопку iCal и скопируйте ссылку в буфер обмена или, если календарь принадлежит кому-то другому, попросите прислать вам эту ссылку. В *Sunbird*, выберите Файл > Создать календарь [File > New Calendar], затем отметьте опцию В сети [On The Network]. Укажите формат iCal и вставьте ссылку на календарь в адресную строку.

Для синхронизации *Sunbird* с Google нужно сделать Google сервером, а *Sunbird* клиентом. То есть нужно сначала установить календарь в Google, а потом уж в *Sunbird*, или воспользоваться уже настроенным календарем Google, если такой у вас есть. Потребуется также установить модуль *Provider* – его можно скачать с сайта **https://addons.mozilla.org/en-US/thunderbird/addon/4631**: он позволяет делать синхронизацию в обоих направлениях. И хотя уже доступна 0.4 версия модуля, вам, возможно, больше подойдет более старая версия 0.3, если у вас установлен *Sunbird 0.7* с *Synaptic*. *Provider 0.4* требует *Sunbird* версии 0.8, который можно скачать с сайта **www.mozilla.org/projects/calendar/sunbird** и установить вручную. Видимо, потребуется установить еще и *libstdc++*, это можно сделать через *Synaptic*.

Как и с подписным календарем, нам снова потребуется уже существующий календарь Google: он у нас будет сервером. Кликните на стрелке около календаря Google, с которым хотите синхронизироваться, выберите Установки календаря, и в Персональном адресе [Private Address] щелкните правой кнопкой мыши по оранжевой кнопке XML и скопируйте адрес ссылки. Теперь перейдите в *Sunbird*, откройте Файл > Создать календарь > В сети и выберите Календарь Google. Вставьте ссылку на XML в адресную строку, а затем введите имя пользователя и пароль Google Calendar. После этого два календаря будут связаны таким образом, что информация, введенная в одном из них, автоматически добавится во второй. Проделайте эту процедуру на всех компьютерах, где используется тот же самый календарь. »



› После установки нажмите на кнопку календаря внизу справа окна сообщения – календарь переключится в полный режим.

› Сервис GooSync позволяет синхронизировать несколько календарей Google с вашим мобильником. Каждому календарю можно присвоить короткий тэг для его идентификации.

Скорая помощь

Provider и *GCALDaemon* позволяют синхронизировать ваш календарь с несколькими календарями Google, так что можно завести события разного рода: работа, дни рождения, частные события. Вы можете выбрать синхронизацию со всеми календарями или только с одним.

Часть 2 Evolution

Если для работы с календарем вы предпочитаете использовать *Evolution*, то существует и другой способ синхронизации с Google Calendar, хотя и более сложный. Тут вам нужен не модуль расширения, а сторонний инструмент под названием *GCALDaemon*, который можно скачать с **http://gcaldaemon.sourceforge.net**. У кого еще не установлена среда Java 1.5, сделайте это сейчас.

Скачав Zip-файл, разархивируйте его в папку **/usr/local/sbin**. Мы пользовались файловым менеджером в Ubuntu; при установке из командной строки, запустите следующее:

```
sudo bash
cd /usr/local/sbin
unzip /downloads/gcaldaemon-linux-1.x.zip
chmod -R g+w /usr/local/sbin/GCALDaemon
chmod 755 /usr/local/sbin/GCALDaemon/bin/*.sh
```

Закройте окно терминала, запустите *Evolution*, создайте новый календарь и назовите его, например, Google. Закройте *Evolution* и просмотрите в файловом менеджере каталог **/home/yourusername/.evolution/calendar/local**. Там должна появиться новая папка; в моем случае это **1218038 862.11.810.22@simon-laptop**, и здесь находится файл **calendar.ics**, соответствующий новому, только что созданному календарю. Откройте окно терминала и скомандуйте следующее:

```
sudo bash
cd /usr/local/sbin/GCALDeamon/bin
./ config-editor.sh
```

Запустится графический редактор настроек или, если вы предпочитаете работать в любимом текстовом редакторе, откройте файл конфигурации: **/usr/local/sbin/GCALDaemon/conf/gcal-daemon.cfg**. В последнем случае примените инструмент шифрования пароля, чтобы он сохранился как хэш, а не в простом текстовом формате:

```
sudo bash
cd /usr/local/sbin/GCALDaemon/bin
./password-encoder.sh
```

Введите пароль от вашего Google Calendar, затем скопируйте зашифрованный хэш в файл конфигурации.

Синхронизируйте еще разок

Если вы вошли в редактор настроек, вы увидите слева несколько закладок. Нажмите на вкладку Синхронизатор файлов [File Synchroniser] и, справа внизу, на Учетные записи Google [Google Accounts]. Тут вам нужно добавить имя пользователя и пароль от вашей учетной записи Google Calendar, используя нормальный (незашифрованный) пароль, который требуется набрать дважды. Нажмите на кнопку Проверить [Verify], чтобы убедиться, что *GCALDaemon* может соединиться с вашей учетной записью Google. Для настройки синхронизации нажмите кнопку New. Выберите учетную запись для синхронизации с *Evolution*. В окне iCal, введите полный путь к файлу *Evolution* iCal – нечто вроде **/home/yourusername/.evolution/calendar/local/128038862.11.81.22@simon-laptop/calendar.ics**. Теперь осталось закрыть инструмент настройки и сохранить изменения.

Поскольку в *Evolution* нет синхронизации в реальном времени, изменения, сделанные в вашем Google Calendar, не появятся в *Evolution*, пока вы не выполните синхронизацию двух календарей и перезапустите *Evolution*, и наоборот. Для синхронизации двух календарей, откройте окно терминала и запустите следующее:

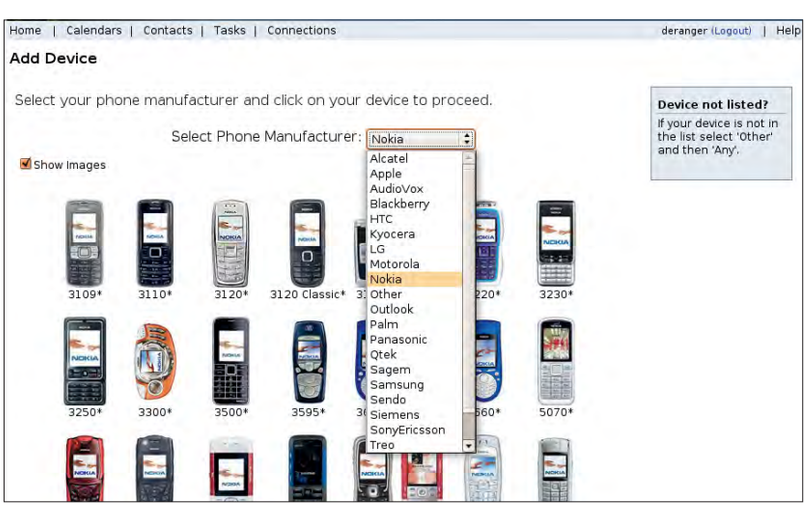
```
sudo bash
cd /usr/local/sbin/GCALDaemon/bin
./standalone-start.sh
```

Чтобы создать новую иконку на рабочем столе – для синхронизации – нажмите правой клавишей мыши и выберите Создать ярлык приложения [Create Launcher]. В поле Тип [Type] укажите Программа в терминале [Application In Terminal], присвойте какое-нибудь имя и введите команду **/usr/local/sbin/GCALDaemon/bin/sync-now.sh**.

Часть 3 Синхронизация с телефоном

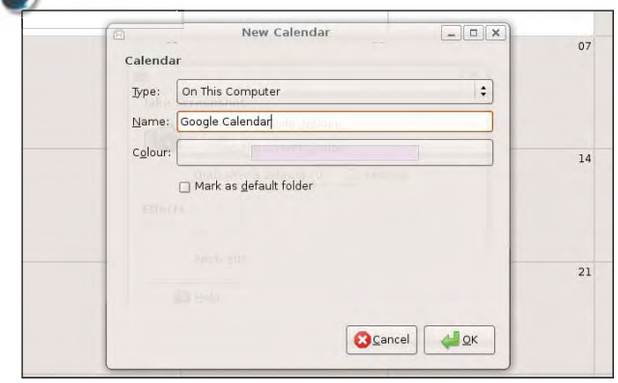
Теперь мы настроили синхронизацию с Google Calendar и для *Evolution*, и для *Sunbird*; но как передать эту информацию на мобильный телефон? Мы обнаружили два способа решения этой задачи: первый – с использованием *GCalSync*, это приложение Java, запускаемое на вашем телефоне. Если ваш телефон может запускать приложения Java через встроенный интернет-обозреватель, скачайте приложение отсюда: **http://wap.gcalsync.com**. Программа бесплатная, но не лишена ограничений и известных проблем: изучите **www.gcalsync.com** для более подробной информации. Кому так удобнее, можно скачать файлы с главного сайта, а потом передать их на телефон вручную. Существует подписанная и неподписанная версии, на случай, если телефон не примет первую.

› GooSync поддерживает огромное количество телефонов от разных изготовителей.



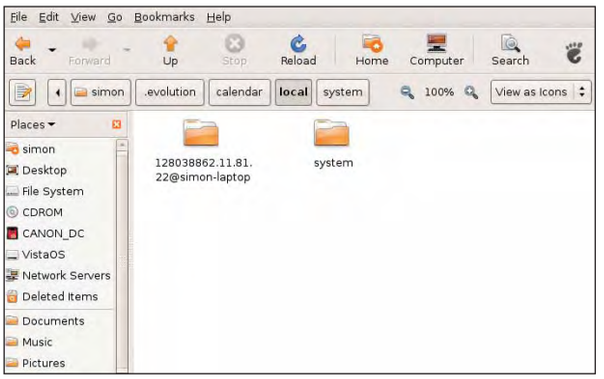
Доступен и более быстрый способ, но, к сожалению, не бесплатный – *GooSync*. Он работает со множеством телефонов и использует встроенный сервис синхронизации – у большинства телефонов, имеющих функцию календаря, он есть. Можно синхронизироваться с несколькими Календарями Google (на одной учетной записи), и каждое событие будет выводиться с коротким кодом-префиксом, чтобы было понятно, какому календарю оно принадлежит. Кроме синхронизации с вашими календарями, *GooSync* умеет также создавать резервную копию ваших контактов – те, кто хоть раз терял свой телефон, весьма оценят эту полезную функцию. *GooSync* стоит около 1000 рублей в год, а пожизненную подписку можно приобрести за 2000. Вдобавок можно создать несколько папок контактов – так вы сможете сохранить больше контактов, чем помещается в телефоне, и дописывать их либо удалять из папки синхронизации, или перенести устаревшие контакты в другую папку, если вы не хотите их забывать, но сейчас они вам в телефоне не требуются.

Шаг за шагом: Настройка GCALDaemon



1 Начнем

Запустите *Evolution* и создайте новый календарь, выбрав Создать > Календарь [New > Calendar]. Выберите создание календаря на вашем компьютере и дайте ему имя, например, Google Calendar. Выберите цвет, который вы хотите использовать для событий.

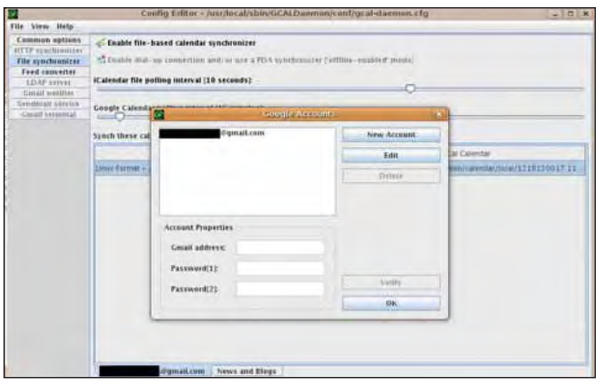


2 Найдите правильный файл

Используя файловый менеджер, включите опцию показа скрытых файлов в меню Вид [View], затем выберите **/home/username/.evolution/calendar/local**. Запишите имя только что созданной папки, которая содержит файл нового календаря.

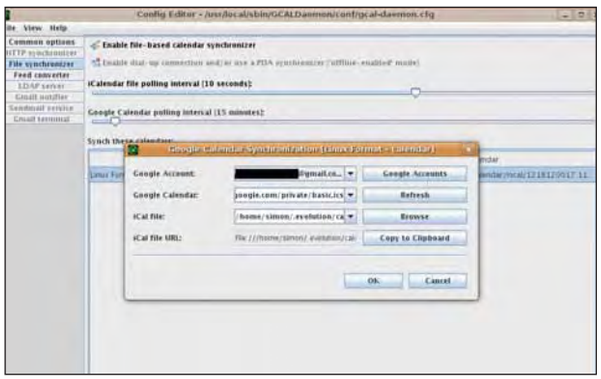
3 Графическая настройка

Откройте окно терминала, перейдите в каталог **binaries** программы *GCALDaemon* и запустите **config-editor.sh**. После запуска GUI, выберите нужные функции – например, оповещение – из вкладки Стандартные параметры [Common Options].



4 Давай подробности

Для добавления параметров учетной записи Google откройте вкладку Синхронизатор файлов, затем нажмите на кнопку Google Accounts внизу слева. Введите ваш email адрес, имя пользователя и – дважды – пароль. Нажмите на кнопку Verify для проверки, что *GCALDaemon* может соединиться с вашей учетной записью.



5 Путь к успеху

Нажмите на кнопку Создать и добавьте требуемые параметры: вашу учетную запись Google, Google Calendar, с которым вы хотите синхронизироваться, и путь к ICS-файлу в *Evolution*, созданному на шаге 1. Указав все детали, закройте редактор конфигурации и при подсказке сохраните изменения.

6 Начните синхронизацию

До запуска *Evolution* нужно запустить *GCALDaemon* – откройте окно терминала, перейдите туда, где хранятся двоичные файлы *GCALDaemon* и запустите **standalone-start.sh**. Если все выполнено правильно, вы получите список статусных уведомлений. Если в нем нет ошибок, синхронизация работает, и можно запускать *Evolution*.

Nagios: Следим за

Большой Брат наблюдает за каждым нашим шагом; **Джульетта Кемп** покажет, как уделить такое же внимание локальной сети с помощью Nagios.

Nagios предоставляет небывалый контроль за всеми компьютерами сети, позволяющий исправить ошибки до того, как они станут критическими, и быть уверенными в том, что вы знаете о своей сети все. Базовая структура *Nagios* очень проста. Один из компьютеров становится сервером *Nagios*, собирает информацию с указанных клиентских компьютеров и аккуратно отображает ее на web-странице. Для получения базовых сведений о клиентских компьютерах (пингуются ли они, и работают ли SSH и HTTP) на них не нужно устанавливать никакого ПО: всю работу сделает сервер. Для получения централизованных отчетов с дополнительной информацией (например, доступным дисковым пространством или использованием процессора) на клиентские компьютеры потребуется установить ПО. При возникновении аварийных ситуаций сервер может реагировать на них путем отправки электронного письма, вывода сообщения на экран или выполнения других действий, заданных в файле скрипта.

Программа чрезвычайно гибкая, настраиваемая и мощная. Оборотной стороной тут неизбежно является сложность правильной настройки. На данном уроке мы рассмотрим конфигурацию центрального сервера и одного клиентского компьютера (все остальные настраиваются точно так же) и слегка коснемся других возможностей, которые можно исследовать после инсталляции. Это непросто, но стоит приложенных усилий, так как после установки Nagios поддержки почти не понадобится.

Наш эксперт

Джульетта Кемп – штатный системный администратор, и она с распростертыми объятиями принимает все, что может уменьшить объем ее работы. *Nagios* ей вполне подходит, спасибо.

Часть 1 Начальная установка сервера

Для начала настроим сервер *Nagios* и просмотр отчетов через web. На сайте www.nagios.org можно найти исходники сервера или установочные пакеты для вашего дистрибутива. Для Debian это пакет *nagios2* (не *nagios*, тот уже устарел); также нужно установить пакет *nagiosplugins* (о модулях расширения мы поговорим позже). Во время установки у вас запросят пароль администратора – не забудьте его! – а также уточнят, нужна ли обратная совместимость с первой версий (для новой установки – не нужна).

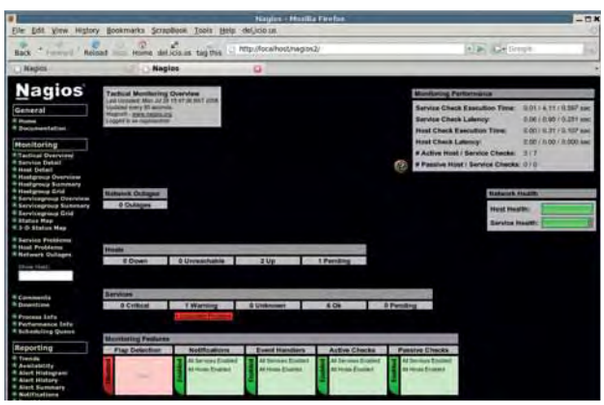
Если у вас еще нет *apache2*, им тоже нужно обзавестись (этого я здесь описывать не буду). Установив *Nagios* и *Apache*, зайдите в каталог **/etc/nagios2** и скопируйте секции из файла **apache2.conf** в конфигурационный файл вашего сервера *Apache*. **apache2.conf** содержит базовые настройки, которые должны обеспечить генерацию web-отчетов в стандартной установке *Apache*. Если до того вы настраивали *Apache* или *Nagios* или перемещали какие-то файлы, убедитесь, что всякие директивы и псевдонимы указывают на нужные места. Перезапустите *Apache*, и с web покончено. Естественно, теперь надо настроить сервер, чтобы он выводил полученную информацию в web-страницы.

Примечание: далее предполагается, что все файлы и каталоги находятся в каталоге **/etc/nagios2**, если не оговорено иное.

В случае с Debian – а я советую применять этот подход и тогда, когда вы устанавливаете сервер на другой системе или из исходных текстов – надо создать каталог **conf.d** и разместить в нем большинство файлов конфигурации. Внутри него, настройки можно раскидать по любому числу отдельных файлов. По-моему, такими файлами управлять проще, чем одним большим, хотя при желании можно избрать и такой путь!

Путь к этому каталогу указывается в основном файле конфигурации **nagios.cfg** следующим образом:

```
cfg_dir=/etc/nagios2/conf.d
```



Экран *Nagios* во время начальной установки.

Месяц назад Мы презрели *OOo* и *Scribus* ради *LaTeX* – 133 способа ввести стили в текст.

своей сетью

Точно так же можно добавить и другие каталоги – столько, сколько понадобится.

Для базовой установки, когда осуществляется мониторинг только самого сервера (**localhost**), должны подойти большинство настроек по умолчанию. Нужно только отредактировать файл **conf.d/contacts_nagios2.cfg**, записав свой адрес электронной почты в первый контакт. При возникновении аварийной ситуации *Nagios* отправит вам письмо. Перед запуском *Nagios* взгляните на настройки по умолчанию для **localhost**, заданные в файле **conf.d/localhost_nagios2.cfg**. Настройки сервера и первого сервиса должны выглядеть примерно так:

```
define host{
use generic-host
host_name localhost
alias localhost
address 127.0.0.1
}
# Сервис, следящий за свободным местом на корневом разделе
# Предупреждение при <20%, критическая ошибка – при <10%
define service{
# Используемый шаблон
use generic-service
host_name localhost
service_description Disk Space
check_command check_all_disks!20%!10%
}
```

Затем определяются остальные сервисы.

Шаблоны

Ключевое слово use в этом фрагменте кода подчеркивает очень удобную функцию *Nagios* – шаблоны. Это означает, что настройки сервиса или основного компьютера можно сохранить и не вводить их каждый раз заново. Повторное использование конфигурации, как и любого другого кода, всегда лучший выбор – меньше мороки, проще поддержка!

Шаблон generic-host находится в файле **conf.d/generic-host_nagios2.cfg**. Он содержит набор установок, в том числе активацию оповещений, различные аспекты обработки оповещений и событий и пр. Однако в настройках индивидуального компьютера любую из них можно изменить. Таким же образом шаблон generic-service задает настройки сервисов (взглянуть на них можно в файле **conf.d/generic-service_nagios2.cfg**). Ключевое слово check_command ссылается на команду, находящуюся в файле **/etc/nagios-plugins/config/disk.cfg**. Данный каталог содержит команды для проверки различных сервисов, и кроме модулей, установленных из пакета по умолчанию (в случае инсталляции из исходников большинство из них придется добавить вручную), можно заполучить огромное количество других.

Пока оставьте основные настройки без изменений. Чуть позже мы добавим другой компьютер в несколько групп. Перезапустите *Nagios* и откройте страницу **http://server.example.com/nagios2** в браузере (потребуется ввести пароль пользователя nagiosadmin, который вы задали при установке). Появится стартовая страница; кликните по ссылке Tactical Overview в меню слева, и появится отчет как на экранном снимке на следующей странице.

В тактическом обзоре содержится информация о компьютерах, сервисах и их состоянии. С точки зрения *Nagios*, компьютеры и сервисы могут быть в одном из нескольких состояний. Pending означает, что проверка состояния еще не сделана; OK – вы удивитесь! – что все хорошо. Для «нехороших» состояний есть несколько уровней: желтый цвет – Предупреждение (Warning), а красный – Критическая ошибка (Critical). Уровни можно установить самостоятельно в файлах конфигурации. При щелчке на сервисе появится окно с подробной информацией о нем (или компьютере). »

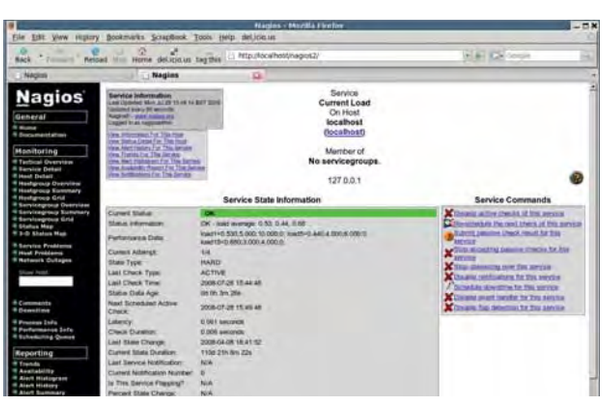
Изменяем стартовую страницу

Чтобы стартовой точкой web-интерфейса Nagios был тактический обзор, а не домашняя страница, отредактируйте файл /usr/share/nagios2/htdocs/index.html, заменив строку

```
FRAME SRC="main.html" NAME="main">
```

на

```
<FRAME SRC="/cgi-bin/nagios2/tac.cgi" NAME="main">
```



› Вот отчет *Nagios* о благополучном сервисе…

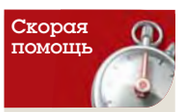
› … а вот – о менее благополучном, в состоянии «Предупреждение».

Часть 2 Компьютеры и сервисы

» Пока сервер *Nagios* осуществляет мониторинг самого себя и шлюза по умолчанию. Теперь нужно добавить клиентский компьютер. Лучший способ сделать это – создать файл **conf.d/host-client1.cfg**, названный по имени компьютера клиента, который сначала должен выглядеть примерно так:

```
define host{
use generic-host
host_name client1
address 10.0.0.2
}
```

Как видите, это самые базовые настройки – почти вся информация берется из шаблона для клиентского компьютера. Перезагрузите *Nagios* (**/etc/init.d/nagios2 reload**), дайте ему минуту-другую на различные проверки и просмотрите отчет.



Ситуация, когда состояние сервиса или компьютера изменяется слишком часто, приводя к появлению множества сообщений об ошибках, называется «переброски». Это может означать неправильную настройку или реальные проблемы с сетью.

Группы компьютеров и сервисы

Теперь *Nagios* может осуществлять мониторинг клиентского компьютера, но пока что для него не определено никаких проверок. Их можно добавить в файл конфигурации этого компьютера точно так же, как и в файл для **localhost**, рассмотренный ранее. Однако делать это для каждого компьютера довольно утомительно – как и с шаблонами, неплохо бы набрать настройки только однажды. Для этого потребуется определить группу компьютеров и затем определить сервис для нее, а не для каждого отдельного хоста.

Отредактируйте файл **conf.d/hostgroups_nagios2.cfg**, добавив имя компьютера в соответствующие группы. При желании можно создать новую группу. Например, у меня есть такая группа:

```
define hostgroup {
hostgroup_name debian-servers
alias Debian GNU/Linux Servers
members localhost,webserver,ldapserver
}
```

Для группы можно задать множество других настроек – здесь поможет отличная страница документации *Nagios*. Однако и эти минимальные настройки отлично сработают.

Итак, клиентский компьютер добавлен в группу; назовем ее **debian-servers**. Отредактируем файл **conf.d/services_nagios2.cfg**, чтобы задать проверки сервисов для этой группы. Проверим «пингуемость» компьютера и состояние SSH:

```
define service {
hostgroup_name debian-servers
service_description SSH
check_command check_ssh
use generic-service
notification_interval 0
}
define service {
hostgroup_name debian-servers
service_description PING
check_command check_ping!100.0,20%!500.0,60%
use generic-service
notification_interval 0
}
```

Возможно, вы захотите определить группу компьютеров, к которым должен быть доступ по SSH, и другую группу компьютеров, которые должны «пинговаться». Например, шлюз по умолчанию пингуется, но не обязательно доступен по SSH. Но сейчас мы определим только одну группу, так как у нас есть всего один клиентский компьютер. *Nagios* проверит компьютеры в этих группах на предмет состояния сервисов, определенных выражением check_command с настройками generic-service, и сообщит о найденных ошибках.

Проверка нескольких сайтов

Если вы хотите проверить, что web-сервер отвечает на HTTP-запросы, отлично подойдет существующая команда check_http. Однако на одном сервере может быть несколько доменов, и проверять их нужно по отдельности. Для этого сначала добавьте в файл **commands.cfg** следующие строки:

```
define command{
command_name check_http-website1
command_line /usr/lib/nagios/plugins/check_http -H website1.
example.com
}
```

Создайте такие же команды для всех сайтов. Затем отредактируйте файл настройки web-сервера (например, **conf.d/host-webserver.cfg**), включив сервис для каждой команды:

```
define service{
host_name webserver
service_description website1
check_command check_http-website1
use generic-service
notification_interval 1440
```

Часть 3 Аварийные ситуации

Ну вот, *Nagios* настроен так, что можно отслеживать состояние компьютеров с одной web-страницы. Теперь добавим отправку электронного письма при каждом возникновении аварийной ситуации, чтобы не открывать браузер попусту. Мы уже редактировали файл **contacts_nagios2.cfg** – в нем также определена группа контактов admin. Опять же, лучше создать группы контактов, чем работать с отдельными пользователями: это упрощает обслуживание системы. Если потребуется изменить настройки пользователей, достаточно будет подправить их членство в группах, а не искать вхождения во всех файлах. По умолчанию группа контактов admins содержит только пользователя root, настройки для которого мы уже задали, так что с ним и продолжим.

Общие настройки для сервиса снова нам помогут. Моя настройка по умолчанию для всего – при любой проблеме отправлять электронное письмо, потому что я проверяю почту довольно часто и скорее найду это сообщение там, чем вспомню, что нужно просмотреть web-страницу. Так мы и сделаем. Добавьте следующее определение сервиса в файл **conf.d/generic-service_nagios2.cfg**:

```
notification_interval 1440
is_volatile 0
check_period 24x7
normal_check_interval 5
retry_check_interval 1
max_check_attempts 10
notification_period 24x7
notification_options c,r
contact_groups admins
}
```

Интервал оповещения определяет, как часто вам будет приходить напоминание (в минутах) – здесь это 24 часа. Временные интервалы определяются в файле **conf.d/timeperiods_nagios2.cfg**. Параметр check_period определяет, когда сервис должен выполняться – в данном случае, все время. Параметры normal_check_interval и retry_check_interval определяются в минутах: в данном случае, сервис проверяется каждые пять минут, но если ответ не получен и выполняется повторная про-

верка, она будет происходить раз в минуту. После десяти повторных попыток *Nagios* делает вывод, что с сервисом что-то не так, хотя можно, если надо, уменьшить этот параметр.

Параметр notification_period определяет, когда должны рассылаться оповещения – опять же, в нашем случае, всегда – а параметр notification_options определяет, при каких условиях генерируется оповещение. Для хостов, d означает оповещение в том случае, если система отключилась, u – если она недоступна, r – сообщать о восстановлении системы, а f – когда на компьютере начинаются и прекращаются переброски. Для сервисов w означает оповещение о состояниях «предупреждение», u – о неизвестных состояниях, c – о критических состояниях, и, опять же, r – восстановление, а f – начало/окончание переброс ок. Наконец, параметр contact_groups определяет, с кем нужно связаться на случай ошибки.

Настроив все это, перезагрузите *Nagios*, затем попробуйте отключить SSH на компьютере клиента. Вы должны получить сообщение, что SSH на клиенте недоступен, на адрес, указанный в файле контактов. Включите SSH – и должно придти другое сообщение, о том, что SSH снова в порядке.

Настройка поля From

По умолчанию в поле From: отправляемого оповещения помещается пользователь *Nagios*. Это может быть не очень хорошо, если почтовому серверу нужен зарегистрированный адрес отправителя. Если вы пользуетесь *Exim 4*, установите опцию 'untrusted user' и добавьте

```
-- -f address@example.com
```

в конец команд **host-notify-by-email** и **notify-by-email** в файле **commands.cfg**.

Часть 4 Модули расширения

Теперь базовый вариант *Nagios* готов, и можно легко добавить еще компьютеры и сервисы, тем же простым способом. Но возможности *Nagios* гораздо богаче.

В качестве примера рассмотрим модуль расширения, позволяющий отслеживать дисковое пространство, использование процессора и другие подобные параметры удаленных компьютеров. Пока вы можете только видеть, включен компьютер или нет. По идее, нужно бы знать больше – вдруг переполнится диск или откажет почтовая служба.

Для этого нам понадобится модуль NRPE. Установите его (в Debian он находится в пакете **nagios-nrpe-plugin**) на сервер *Nagios* и добавьте сервер NRPE на удаленный компьютер (пакет **nagios-nrpe-server** в Debian). Сервер NRPE будет собирать информацию с компьютера и передавать ее модулю, когда с ним свяжется основной сервер.

Проверим, что это работает…

Для проверки соединения между сервером и клиентом, выполните команду /usr/lib/nagios/plugins/check_nrpe -H client -c check_users на сервере: она должна сообщить вам, сколько пользователей зашли в систему на клиентском компьютере. Затем проверьте содержимое файла **/etc/nagios-plugins/config/check_nrpe.cfg** на сервере, если это необходимо – в Debian нужные настройки уже заданы, и редактировать его не нужно. Он должен выглядеть примерно так:

```
define command {
command_name check_nrpe
command_line /usr/lib/nagios/plugins/check_nrpe -H
 $HOSTADDRESS$ -c
$ARG1$ -a $ARG2$
}
 # эта команда запускает программу $ARG1$ с единственным
аргументом
define command {
command_name check_nrpe_1arg
command_line /usr/lib/nagios/plugins/check_nrpe -H
 $HOSTADDRESS$ -c
$ARG1$
}
```

Отредактируйте файл **conf.d/services_nagios2.cfg** на удаленном компьютере, добавив сервисы, которые нужно отслеживать. Мои настройки выглядят так:

```
define service {
service_description SMTP
use generic-service
hostgroup_name nrpe
check_command check_nrpe_1arg!check_smtp
}
define service {
service_description LOAD
use generic-service
hostgroup_name nrpe
check_command check_nrpe_1arg!check_load
}
define service {
service_description DISK
use generic-service
hostgroup_name nrpe
check_command check_nrpe!check_disk!/
}
```

(Учтите, что должна быть настроена группа компьютеров nrpe.)

Сервисы, требующие только один аргумент – имя отслеживаемой службы – используют команду check_nrpe_1arg (см. **/etc/nagiosplugins/config/check_nrpe.cfg**). Если нужно передать дополнительные аргументы, потребуется отредактировать файл **/etc/nagios/nrpe.cfg** на компьютере клиента, задав параметру dont_blame_nrpe значение 1. Затем воспользуйтесь командой check_nrpe. В приведенном фрагменте кода я использовала аргумент, чтобы сообщить точку монтирования проверяемого диска. Перезапустите *Nagios*, и должны появиться данные с клиентского компьютера.

Доступные команды находятся в каталоге **/usr/lib/nagios/plugins** на клиентском компьютер, а в файле **/etc/nagios/nrpe-local.cfg** можно создать собственные. Я создала несколько аргументов, которые выглядят так:

```
command[check_disk]=/usr/lib/nagios/plugins/check_disk -w 10% -c 5%
-p $ARG1$
command[check_smtp]=/usr/lib/nagios/plugins/check_smtp -w 1 -c 2
```

Это означает, что я могу проверить все локальные диски, какие хочу, не ограничиваясь корневым каталогом /, как в случае с командой check_disk по умолчанию, и могу убедиться, что на всех компьютерах благополучно работает SMTP. Точно так же можно создать другие команды. (Кстати, обратите внимание, что команда check_disk выдает количество свободного – а не используемого – пространства на диске).

Итак, наша первая настройка *Nagios* закончена – сервер проверяет себя и удаленный компьютер и сообщает обо всех проблемах. Добавить дополнительные компьютеры и сервисы несложно; если хотите сделать нечто большее, воспользуйтесь каталогом модулей расширения. Команда /usr/lib/nagios/plugins/plugin_name -h выведет справку по указанному модулю. А пока – усядьтесь в кресле поудобнее и наслаждайтесь созерцанием того, как сеть наблюдает сама за собой! LXF

Скорая помощь

Модули расширения позволяют легко обогатить функциональность *Nagios* – найти их можно в репозитории по адресу http://nagiosplugins.org.

Через месяц Чудеса разбиения дисков, совершаемые Менеджером логических томов (LVM).

Раздвигая горизонты

ЧАСТЬ 3 CMake – не только мощный, но и расширяемый инструмент для сборки ваших приложений. **Андрей Боровский** покажет, как добавить ему функциональности.

Сегодня мы завершим знакомство с CMake обзором тех возможностей пакета, которым не нашлось места на предыдущих уроках. Из этой статьи вы узнаете, как писать собственные сценарии расширения CMake, научитесь устанавливать ПО и создавать дистрибутивы (нет, не Linux, но тоже полезные).

Нестандартные связи

Давайте условимся понимать под стандартными библиотеками те, для которых уже имеются готовые модули (сценарии) расширения CMake. Если вы пишете новую библиотеку, вполне логично создать для нее и модуль расширения. Этим мы займемся позже, а сейчас рассмотрим действия, необходимые для подключения библиотеки к проекту. Прежде всего, нам потребуется нестандартная (в указанном выше смысле) библиотека.

Структурно проект библиотеки мало чем отличается от проекта программы. Фактически, вся разница сводится к одной-единственной команде. На прилагаемом диске вы найдете проект библиотеки *demolib*, которая экспортирует функцию testfunc(). Последняя, в свою очередь, распечатывает на экране сообщение о своем вызове (вряд ли в природе существует более простая библиотека). Исходный текст нас сейчас не интересует, так что перейдем сразу к файлу **CMakeLists.txt**:

```
cmake_minimum_required(VERSION 2.6)
project(demolib C)
add_library(demolib SHARED demolib.c demolib.h)
if(${CMAKE_SYSTEM_NAME} STREQUAL Windows)
set(LIB_INSTALL_PATH ${CMAKE_INSTALL_PREFIX}/lib/)
elseif(${CMAKE_SYSTEM_NAME} STREQUAL Linux)
set(LIB_INSTALL_PATH /usr/lib/)
endif()
install(TARGETS demolib DESTINATION ${LIB_INSTALL_PATH})
find_path(LIB_INCLUDE_PATH string.h)
install(FILES demolib.h DESTINATION ${LIB_INCLUDE_PATH})
```

Значительная часть команд в этом файле предназначена для обеспечения кроссплатформенности. За обычным заголовком мета-проекта *CMake* следует команда add_library(). Как нетрудно догадаться, она представляет собой аналог уже знакомой нам команды add_executable(), только в качестве цели сборки выступает не исполняемый файл программы, а библиотека. Первым аргументом команды add_library() должно быть ее имя (которое по совместительству является именем соответствующей цели сборки). Имя указывается в кроссплатформенном виде (без префикса lib и расширения so). Далее следует тип создаваемой библиотеки (SHARED – разделяемая, STATIC – статическая, MODULE – динамически загружаемый разделяемый модуль [на большинстве систем эквивалентен SHARED, – *прим. ред.*]). Затем, как и в команде add_executable(), мы указываем список файлов исходных текстов, необходимых для сборки цели.

Конструкция if(${CMAKE_SYSTEM_NAME} STREQUAL XXX) позволяет определить, выполняется ли сценарий *CMake* на платформе XXX или на какой-либо другой (LXF110). Допустимо использовать и более краткую запись: if(WIN32), if(UNIX). Переменная UNIX обозначает все Unix-системы, но поскольку Solaris – это все же не HP-UX и не AIX, я предпочитаю более конкретный вариант с проверкой CMAKE_SYSTEM_NAME.

В приведенном выше примере мета-проекта задействована еще одна возможность *CMake*, с которой мы ранее не встречались – установка собранной цели. *CMake* предоставляет несколько команд, с помощью которых можно добавить в создаваемый проект средства инсталляции ПО. Самой полезной из них является install(), принимающая три группы аргументов: спецификатор, определяющий, что именно мы устанавливаем, список имен объектов и целевую директорию. Команда install(), вызванная со спецификатором TARGETS, устанавливает файлы, являющиеся целями (т.е. результатом сборки). В качестве аргументов ей передаются имя цели и каталог, в который должны быть помещены файлы. Чтобы сгенерировать инструкции установки файла, не являющегося целью сборки (например, **demolib.h**), используется команда install() со спецификатором FILES. Есть и другие опции: можно, например, указать, какую из конфигураций сборки (RELEASE, DEBUG и т.д.) следует использовать для инсталляции (если вы думаете, что никому не понадобится устанавливать DEBUG-проект, то ошибаетесь: многие библиотеки, модули расширения, да и программы можно отлаживать только после полной установки). В команде install() можно также указывать права доступа для устанавливаемого файла. Спецификатор SCRIPTS команды install() позволяет выполнять сценарии *CMake* до и после установки. Это может оказаться полезным в тех случаях, когда для корректной инсталляции необходимо не только скопировать файл, но и выполнить некоторые дополнительные действия – запустить утилиты, настроить файлы конфигурации, добавить записи в реестр (ой, о чем это я?..).

Корректная установка файлов возможна только при правильном выборе целевых директорий. В Linux и других *nix проблем обычно не возникает (таким образом, мы еще раз убеждаемся в технической рациональности идеи файловой системы с единым корнем). На платформе Windows все гораздо сложнее. Мало того, что разные важные директории (точнее – каталоги) могут быть расположены на разных дисках; в Windows вообще не существует единых правил относительно установки библиотек и разделяемых файлов. Например, популярные динамические библиотеки копируются в каталоги **%WINDIR%/system/**, **%WINDIR%/system32/** и т.п., однако это касается только DLL; lib-файлы, необходимые для компоновки программ с разделяемыми библиотеками, устанавливаются в директории сред разработки. В то же время, команда install() по умолчанию копирует DLL- и LIB-файлы в один каталог.

В нашем мета-проекте мы сохраняем полное имя директории для установки библиотеки в переменной LIB_INSTALL_PATH. На

Месяц назад Мы разбирались, как извлечь информацию о системе и что с ней потом делать.

Подключение библиотеки

Простой библиотеке – простая программа. На диске вы найдете приложение *libtests*, которое вызывает функцию testfunc() из библиотеки *demolib*. Само подключение выполняется с помощью уже знакомой нам команды target_link_libraries(). Ниже следует файл **CMakeLists.txt** для сборки программы *libtest*.

```
cmake_minimum_required(VERSION 2.6)
project(libtest)
find_path(DEMOLIB_INCLUDE_DIR demolib.h)
include_directories(${DEMOLIB_INCLUDE_DIR})
add_executable(libtest libtest.c)
if(${CMAKE_SYSTEM_NAME} STREQUAL Windows)
target_link_libraries(libtest $ENV{PROGRAMFILES}/demolib/lib/demolib.lib)
elseif(${CMAKE_SYSTEM_NAME} STREQUAL Linux)
target_link_libraries(libtest demolib)
endif()
```

платформе Windows мы записываем сюда значение ${CMAKE_INSTALL_PREFIX}/lib/ (которое разрешается, например, в **C:\Program Files\demolib\lib**), а под Linux используем жестко заданное **/usr/lib/**. Обратите внимание на важную особенность работы install(): если указанная в этой команде директория не существует, она будет создана. С одной стороны, это удобно, с другой – в случае опечатки в системе могут появиться странные каталоги. В ходе моих экспериментов с Windows и Linux были случайно созданы директории **C:\usr\local\lib** и **/usr/lib/;/**.

Для определения имени директории, в которую следует установить заголовочный файл, мы пользуемся довольно распространенным при работе в *CMake* методом интроспекции (LXF110): с помощью команды find_path() определяем каталог, в котором располагается какой-либо общераспространенный файл того же типа (в нашем примере – **string.h**), и устанавливаем **demiolib.h** в него же. В Linux искомой директорией почти непременно окажется **/usr/include/** (можно было бы и не напрягаться), а вот при работе под Windows все будет сложнее. Путь к директории заголовочных файлов зависит от того, какое средство разработки мы используем и где оно установлено. Замечено, что под Windows данный метод интроспекции не всегда может корректно определить требуемый каталог с первого раза – нужно выйти из программы *CMake GUI* и запустить ее снова. Альтернативный вариант – указать расположение заголовочного файла вручную в окне CMake GUI (само наличие такой опции является признанием того, что интроспекция под Windows работает хуже, чем хотелось бы).

Как изменится сгенерированный проект от того, что в метапроект была добавлена команда install()? На каждой ОС это будет выглядеть по-своему. Если целевой платформой является Linux, в make-файл добавляется цель install, так что установить нашу библиотеку можно командой

```
sudo make install
```

При работе под Windows (среда Microsoft Visual C++) в решение (Solution) Visual Studio добавляется специальный проект INSTALL. Выглядит это несколько неуклюже, но другого универсального способа установки проектов под Windows, по-видимому, пока что не существует.

Создание модуля CMake

Если разделяемую библиотеку планируется использовать во многих проектах, целесообразно написать для нее собственный модуль, в котором будет выполняться поиск связанных с библиотекой файлов. Мы проделаем все это для библиотеки *demolib* (хотя, честно говоря, ее широкое распространение не предвидится). О том, что именно делают модули расширений *CMake*, говорилось в LXF110, поэтому перейдем сразу к начинке (файл **FindDemoLib.cmake**):

```
include(FindPackageHandleStandardArgs)
if(DEMOLIB_INCLUDE_DIR AND DEMOLIB_LIBRARIES)
set(DemoLib_FIND_QUIETLY TRUE)
endif(DEMOLIB_INCLUDE_DIR AND DEMOLIB_LIBRARIES)
find_path(DEMOLIB_INCLUDE_HINT string.h)
find_path(DEMOLIB_INCLUDE_DIR demolib.h HINTS ${DEMOLIB_INCLUDE_HINT})
find_library(DEMOLIB_LIBRARIES demolib HINTS $ENV{PROGRAMFILES}/demolib/lib/ /usr/lib)
find_package_handle_standard_args(DemoLib DEFAULT_MSG DEMOLIB_LIBRARIES DEMOLIB_INCLUDE_DIR)
mark_as_advanced(DEMOLIB_LIBRARIES)
```

Минимум, что должен делать модуль загрузки разделяемой библиотеки *XXX* – это записывать в переменные XXX_INCLUDE_DIR и XXX_LIBRARIES пути к заголовочным файлам и самой библиотеке, соответственно. Кроме того, должны быть инициализированы служебные переменные, например, XXX_FOUND. Вдобавок модуль может предоставлять специальные команды, дополнительные переменные и многое другое; но в нашем примере мы ограничимся малым. Модуль FindDemoLib записывает путь к библиотеке *demolib* в переменную DEMOLIB_LIBRARIES, а путь к файлу **demolib.h** – в переменную DEMOLIB_INCLUDE_DIR.

В первой строчке **FindDemoLib.cmake** мы загружаем модуль FindPackageHandleStandardArgs, который содержит полезную вспомогательную команду. Далее мы проверяем, не установлены ли уже значения переменных DEMOLIB_INCLUDE_DIR и DEMOLIB_LIBRARIES. Если обе переменные инициализированы, значит, они уже присутствуют в кэше (LXF110). Если кэш обновлять не нужно, мы присваиваем значение TRUE переменной DemoLib_FIND_QUIETLY.

Обратите внимание на префикс DemoLib, который соответствует основе имени файла модуля. В процессе обработки модуля система *CMake* проверяет значение этой и еще нескольких подобных служебных переменных. Дальше мы выполняем ту самую интроспекцию, ради которой все и затевалось. В команде find_path() используется новый элемент – спецификатор HINTS. Он позволяет нам делать среде *CMake* «подсказки», упрощающие поиск файлов. За спецификатором HINTS обычно следует список директорий, в которых может находиться (а может и не находиться) искомый файл. Если он не будет найден в «подсказанных» директориях, система выполнит стандартный поиск. Спецификатор HINTS не следует путать со спецификатором PATHS, с помощью которого мы можем жестко указать список директорий для поиска. Отметим, что даже с подсказкой система не всегда может найти директорию заголовочных файлов на платформе Windows. В этом случае придется вводить имя директории вручную в окне графической утилиты CMake.

Команда find_package_handle_standard_args(), предоставляемая загруженным модулем, выполняет рутинные действия по инициализации служебных переменных. Ее первый аргумент – основа имени модуля, которая используется, например, для генерации имен. Второй аргумент определяет, что именно программа должна сказать в том случае, если библиотека *demolib* не будет найдена. Вместо значения DEFAULT_MSG можно указать свой собственный текст. Далее следуют имена переменных, в которых содержаться путь к библиотеке и заголовочным файлам соответственно.

Завершающая команда mark_as_advanced() помечает переменные, переданные ей в качестве аргумента, как «продвинутые» (advanced). Продвинутые переменные обычно не отображаются в окне графического инструмента CMake.

Файл **FindDemoLib.cmake** следует скопировать в директорию **Modules** (здесь хранятся расширения CMake). В результате в метапроекте для сборки *libtest* мы сможем обойтись без интроспекции:

```
cmake_minimum_required(VERSION 2.6)
project(libtest)
find_package(DemoLib REQUIRED)
include_directories(${DEMOLIB_INCLUDE_DIR})
```

Скорая помощь

Если в метапроекте *CMake* вам нужно получить значение переменной окружения, для которой не существует встроенного «двойника», можно воспользоваться конструкцией $ENV{ИМЯ_ПЕРЕМЕННОЙ}, например:

```
$ENV{SHELL}
$ENV{WINDIR}
```

Синтаксис обращения к переменным окружения из *CMake* не зависит от платформы.

```
add_executable(libtest libtest.c)
target_link_libraries(libtest ${DEMOLIB_LIBRARIES})
```

Подготовка к распространению

Помимо пакета *Cmake*, компания Kitware выпускает еще несколько полезных утилит, в том числе *CPack* – средство создания дистрибутивов. *CPack* входит в состав пакета *Cmake*, и им можно управлять из сценариев *CMake*, так что будет уместно рассмотреть его здесь. Чтобы задействовать *CPack* в сценарии *CMake*, достаточно подгрузить модуль:

```
include(CPack)
```

Если теперь мы запустим утилиту *cmake*, то в результирующем Make-файле появятся цели package и package_source. Первая предназначена для создания двоичного пакета, вторая – для дистрибутива исходных текстов. Если теперь мы наберем

```
sudo make package
```

(эту команду необходимо выполнять от имени суперпользователя), то в результате получим файл сценария оболочки с расширением **.sh**, а также архивы **.tar.gz**, **tar.Z** и **tar.bz2**. Имена файлов сконструированы из имени проекта, номера версии и названия платформы. Например, для проекта *demolib*, на примере которого мы изучаем *CPack*, все они будут называться **demolib-0.1.1-Linux**. Перечисленные файлы представляют собой двоичные пакеты в разных форматах (по умолчанию *CPack* создает сразу несколько пакетов). Файл с расширением **.sh** – это сценарий оболочки с встроенным архивом **tar.gz**. Если мы запустим его на выполнение, он задаст нам несколько вопросов по поводу согласия с лицензией и путей установки, после чего (если наши ответы его устроят) распакует содержимое встроенного архива в заданную директорию. Файлы с расширениями **.tar.*** в комментариях не нуждаются.

Если изложенное выше навело вас на мысль, что создавать двоичные пакеты с помощью *CPack* очень просто, то вы почти правы. На практике, однако, можно столкнуться с некоторыми сложностями. Механизм генерации пакетов *CMake-CPack* использует инструкции, заданные нами для генерации цели install (установки проекта). Попросту говоря, для создания цели package умолчательное значение переменной CMAKE_INSTALL_PREFIX (напомню, что она содержит путь к корневой директории инсталляции) заменяется на путь к некому временному каталогу. Далее вызывается цель install, в результате чего выполняется «холостая» установка проекта во временную директорию, содержимое которой упаковывается в архивы. Этот факт имеет несколько последствий. Во-первых, вы можете создать двоичный пакет только в том случае, если ваш мета-проект содержит инструкции для генерации цели install. Во-вторых, поскольку процесс создания пакета использует подмену значения CMAKE_INSTALL_PREFIX, генерация может пройти успешно лишь в том случае, когда команды install() ее используют. Если вы выполняете нестандартные действия с директориями, будьте готовы к неожиданным проблемам с генерацией пакетов. Чтобы *CMake* мог задействовать переменную CMAKE_INSTALL_PREFIX, в команде install() следует указывать относительные, а не абсолютные пути (например, **lib**, а не **/usr/lib**). Наконец, в-третьих, выполнение процесса установки в ходе генерации пакета может вызвать побочные эффекты в том случае, когда установка включает в себя какие-то действия помимо простого копирования файлов.

Настройка *CPack* из мета-проектов *CMake* выполняется с помощью переменных, которые использует модуль *CPack*. Перечислим наиболее интересные из них:

- CPACK_BINARY_DEB, CPACK_BINARY_RPM, CPACK_BINARY_STGZ – указывают, нужно ли создавать пакет в формате Debian, RPM или **.sh**.
- CPACK_BINARY_TGZ, CPACK_BINARY_TZ, CPACK_BINARY_TBZ2 – управляет созданием архивов **tar.gz**, **tar.Z** ил **tar.bz2**.

Последним четырем переменным по умолчанию присвоено значение ON, первым двум – OFF.

- CPACK_INSTALL_PREFIX – переменная, в которой сохраняется полное имя корневой директории для установки проекта.
- CPACK_PACKAGE_DESCRIPTION_FILE – путь к файлу с развернутым описанием собираемого пакета.
- CPACK_PACKAGE_DESCRIPTION_SUMMARY – краткое описание собираемого пакета.
- CPACK_PACKAGE_FILE_NAME – основа имени файла пакета.
- CPACK_PACKAGE_INSTALL_DIRECTORY – директория, в которую по умолчанию извлекается содержимое пакета.
- CPACK_PACKAGE_VENDOR – имя сборщика пакета.
- CPACK_PACKAGE_VERSION_MAJOR, CPACK_PACKAGE_VERSION_MINOR, CPACK_PACKAGE_VERSION_RELEASE – эти переменные содержат три цифры номера версии распространяемого ПО – старшую, младшую и номер релиза соответственно (используется, в том числе, при конструировании имени файла пакета).
- CPACK_RESOURCE_FILE_LICENSE – путь к файлу с текстом лицензии.
- CPACK_RESOURCE_FILE_README – путь к файлу **README**.
- CPACK_SYSTEM_NAME – имя системы (используется, в том числе, при конструировании имени файла пакета).

Если результатом сборки является скрипт или пакет RPM, информация из файлов лицензии, **README** и WELCOME становится его частью. Чтобы изменить настройки *CPack*, заданные по умолчанию, нужно отредактировать значения соответствующих переменных перед вызовом include(). Например, если мы хотим создать пакет RPM и привести его имя к классическому виду, можно написать:

```
set(CPACK_BINARY_RPM ON)
set(CPACK_SYSTEM_NAME i686)
include (CPack)
```

С учетом всего вышеизложенного вариант сценарий сборки *demolib* с дополнительной целью package выглядит так:

```
cmake_minimum_required(VERSION 2.6)
project(demolib C)
if(UNIX)
set(CMAKE_INSTALL_PREFIX /usr)
set(CPACK_BINARY_RPM ON)
set(CPACK_SYSTEM_NAME i686)
endif(UNIX)
set(CPACK_PACKAGE_DESCRIPTION_SUMMARY "Demo Library Project")
set(CPACK_PACKAGE_VERSION 1.0.0)
include(CPack)
add_library(demolib SHARED demolib.c demolib.h)
install(TARGETS demolib DESTINATION lib)
install(FILES demolib.h DESTINATION include)
```

На платформе Windows для создания двоичных пакетов можно использовать Nullsoft NSIS (что выходит за рамки этой статьи) и zip (что не очень удобно с точки зрения Windows-пользователя).

Как уже говорилось, с помощью *CPack* можно создавать не только двоичные пакеты, содержащие собранное ПО, но и дистрибутивы исходных текстов. По умолчанию, вызов

```
make package_source
```

приводит к тому, что все содержимое корневой директории проекта и всех ее поддиректорий (в том числе, с двоичными файлами) упаковывается в архив. Более того, поскольку сам файл пакета исходников по умолчанию сохраняется в той же корневой директории, может возникнуть ситуация, при которой упаковщик будет пытаться заархивировать файл сам в себя. Управление настройкой генератора пакетов исходных текстов также выполняется с помощью переменных, имена которых начинаются с префикса CPACK_SOURCE_. Как и в случае с *CMake*, вы можете узнать много полезного о переменных *CPack*, ознакомившись с файлами **CPackConfig.cmake** и **CPackSourceConfig.cmake**. Некоторые переменные из этих файлов попадают в кэш *CMake*.

Надеюсь, что после всего сказанного о *CMake* вы придете к тем же выводам, к которым пришел и я – этот пакет не только является средством кроссплатформенной сборки, но и упрощает жизнь программиста, работающего исключительно в Linux. LXF

На всех языках мира

ЧАСТЬ 4 Угадайте, какая тема неизменно замалчивается западными авторами учебников по компьютерным технологиям? Правильно – интернационализация! **Андрей Боровский** спешит заполнить данный пробел.

Кем бы был Джон Рональд Руэл Толкиен, если бы ему довелось дожить до наших дней? Я просто уверен, что Профессор входил бы в команду интернационализации KDE. Чтобы понять это, достаточно прочесть его руководство по переводу топонимов и омонимов для собственных произведений.

На данном уроке мы сосредоточим свое внимание на специфике интернационализации и локализации программ в KDE 4. Система интернационализации KDE 4 основана на стандартном для Linux пакете GNU *gettext* (**www.gnu.org/software/gettext**), так что если вы хотите изучить вопрос более подробно, рекомендую обратиться к его документации.

Если вы выполняли интернационализацию и локализацию в KDE 3.x, то обнаружите, что изменения, привнесенные в KDE 4.x, невелики, но учитывать их надо. Появилась новая среда сборки программ (она теперь основана на CMake), в архитектуру библиотек *Qt*/KDE также были внесены изменения. Да и сама система интернационализации обзавелась новыми функциями. В качестве упражнения мы выполним интернационализацию и русскую локализацию программы, которую создает по умолчанию генератор *kapptemplate*.

Подготовим исходники

Функции и макросы, необходимые для интернационализации KDE-программы, становятся доступны при включении в исходные тексты заголовочного файла **klocale.h**; генератор шаблонов *kapptemplate* делает это автоматически. Основой интернационализации, как и прежде, являются функция i18n() и макрос I18N_NOOP(). Напомню, что функция i18n() получает в качестве параметра строку char * в кодировке UTF-8 и заменяет ее на аналог из каталога переведенных строк. Макрос I18N_NOOP() используется там, где функция i18n() применяться не может (она становится доступна только после создания объекта KApplication). Таким образом, в первом приближении задача подготовки исходных текстов к интернационализации сводится к «оборачиванию» строк английского текста функцией i18n() и макросом I18N_NOOP() (если вы посмотрите исходные тексты программы i18demo, то увидите, что это уже сделано). Далее мы извлекаем из файлов исходных текстов строки, подлежащие переводу, с помощью утилиты *xgettext*, например:

```
xgettext -C -ki18n -kI18N_NOOP *.cpp
```

О терминах

Под интернационализацией мы подразумеваем подготовку программ к переводу на различные языки. Локализация – это собственно перевод на определенный язык (а также адаптация программы к правилам написания чисел и дат, принятым в целевой стране, местной системе мер, используемым денежным единицам и т.п.). Интернационализация выполняется на этапе разработки программы, а локализация может быть выполнена после того, как приложение уже собрано.

Ключ -C указывает команде, что она обрабатывает исходные тексты на C/C++. Параметры -ki18n и -kI18N_NOOP сообщают, что нужно искать строки, обернутые i18n и I18N_NOOP (формат: -k + имя функции или макроса). В результате на диске появляется файл **messages.po**, который содержит все помеченные для перевода строки из файлов с расширением **.cpp**. Затем их нужно перевести на целевые языки и скомпилировать в mo-каталоги. Теперь, чтобы программа «заговорила» на иностранном языке, достаточно просто скопировать эти файлы в нужную директорию.

Однако не все так просто. Наша демонстрационная программа отсчитывает и показывает в специальном окне число дней, прошедших с момента создания проекта. В ситуации, когда программа выводит сообщение о количестве чего-то, мы должны позаботиться о правильном использовании грамматических форм единственного и множественного числа. Допустим, у нас есть фраза вида «прошу выделить мне %1 кочерги», где спецификатор %1 во время выполнения программы заменяется на некое целое число (подробнее о форматировании строк можно прочитать в документации по *Qt*). Можно, конечно, упростить себе жизнь и написать по-канцелярски: «прошу выделить мне изделие “кочерга” в количестве %1 штук». Однако в наше время высоких технологий пользователь вправе ожидать, что компьютеры будут говорить, как люди. Мы хотим, чтобы форма слова «кочерги» менялась в зависимости от количества (1 кочергу, 2 кочерги...). Эту задачу можно решить с помощью кода на C++ и учебника грамматики, но такой подход будет работать только для одной группы языков. Многие программы, обработка множественного числа и других языковых особенностей в которых «зашита» в код, по умолчанию поддерживают только правила английского языка. Эти правила также справедливы для некоторых западноевропейских языков, но не для восточноевропейских, что существенно затрудняет (а иногда и делает невозможной) локализацию программ. Основное правило написания программы, предназначенной для перевода на другие языки: никаких предположений касательно грамматики.

На первый взгляд, задача построения системы интернационализации, которая бы подходила для любого языка, кажется неразрешимой. Тем не менее, разработчики *GNU gettext* с честью справились с этой проблемой, а программисты KDE создали удобные инструменты для ее решения на своей платформе. Для обработки строк, написание которых должно меняться в зависимости от значения целочисленного аргумента, в систему интернационализации KDE добавлена функция i18np(). В исходном в файле i18ndemoview.cpp заменяем строку

```
i18n("This project is %1 days old", Settings::val_time())
```

на

```
i18np("This project is %1 day old", "This project is %1 days old",
Settings::val_time())
```

Кстати, обратите внимание на то, что функции i18n() и i18np() позволяют форматировать строки, заменяя спецификаторы вида %n значениями переменных, подобно методу QString::arg(). У функции i18np() должно быть как минимум три параметра: строка, в которой об исчисляемом

Месяц назад Поставщики данных – механизм, снабжающий информацией апплеты-плазмоиды.

Вредные советы

Некоторые конструкции могут затруднить подготовку программы к переводу на другие языки. Самый распространенный пример того, чего не следует делать – это сборка строки из нескольких кусков: i18n("Time left is") + seconds + i18n("seconds"). В таком варианте функция i18n() не будет правильно работать с множественным числом. В программе, предполагающей перевод на другие языки, не следует рассчитывать на то, что текст всегда будет выводится слева направо, и уж конечно, не следует проектировать интерфейсы, исходя из длины строк в пикселях.

предмете говорится в единственном числе, строка, в которой исчисляемый предмет упомянут во множественном числе, и целочисленная переменная, играющая роль управляющего числительного. Проницательный читатель заметит, что описанный синтаксис вызова i18np() предполагает использование всего лишь одной формы для множественного числа, тогда как в русском языке их больше. На самом деле беспокоиться не о чем. Интерфейс программы мы пишем по-английски, и вполне естественно, что мы ориентируемся на грамматику английского языка. Особенности русской грамматики будут учтены в ходе локализации.

Сосчитай до ста

Мы подготовили исходный текст программы для интернационализации и можем вызвать утилиту *xgettext* для генерации каталога фраз, предназначенных для перевода. Теперь вызов *xgettext* будет выглядеть так:

```
xgettext -C -ki18n -kI18N_NOOP -ki18np:1,2 *.cpp
```

В результате на диске появится файл **messages.po** (имя можно изменить с помощью ключа -o). Если мы откроем его в текстовом редакторе, то, помимо прочего, увидим в нем следующее:

```
msgid "This project is 1 day old"
msgid_plural "This project is %1 days old"
```

то есть в каталог сообщений добавлены обе формы. Для перевода строк на русский язык файл **messages.po** придется «обработать напильником». Найдите в заголовке файла строку

```
"Plural-Forms: nplurals=INTEGER; plural=EXPRESSION;\n"
```

и замените ее на

```
"Plural-Forms: nplurals=3; plural=(n%10==1 && n%100!=11 ? 0 : n%10>=2
&& n%10<=4 && (n%100<10 || n%100>=20) ? 1 : 2);\n"
```

Здесь мы указываем, что общее количество форм множественного числа равно трем (значение параметра nplurals), и присваиваем параметру plural однострочное выражение на C, которое позволяет выбрать нужную форму в зависимости от значения управляющей переменной. Это выражение взято из документа, который можно найти по ссылке www.gnu.org/software/automake/manual/gettext/Plural-forms.html. Там же приведены аналогичные выражения для других языков. Обратите внимание на последовательность форм, которая задается выражением plural. Первая форма (индекс 0) соответствует единственному числу и всем числительным, заканчивающимся на 1 (см. врезку). Вторая форма (индекс 1) – числительным, оканчивающимся на 4–9 и 11–19, третья форма – всем остальным числительным.

Особенности национальной грамматики

О сочетании существительных и числительных в русском языке написаны целые трактаты, и если вы думаете, что знаете об этом все, то наверняка ошибаетесь (подумайте, например, как правильно сочетаются существительное «сутки» и числительное «двадцать два»). В большинстве случаев мы пользуемся тем фактом, что управляемое числительным имя существительное может находиться в одной из трех форм. Первая форма – для единственного числа и всех числительных, заканчивающихся на «один», вторая форма – для всех числительных, оканчивающихся на «два», «три» и «четыре», третья форма – для всех числительных, оканчивающихся на «пять», «шесть», «семь», «восемь», «девять», а так же для числительных, содержащих в двух младших разрядах 11–19, и всех числительных, соответствующих цифрам, у которых в младшем разряде находится 0 (сами формы для разных существительных выглядят по-разному, но с этим у читателей русского LXF не должно возникнуть проблем).

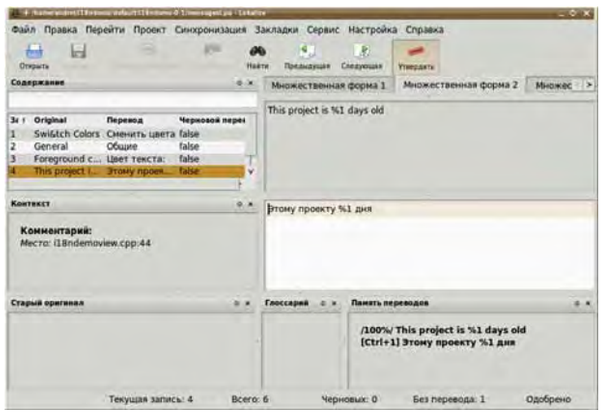


Рис. 1. Программа *Lokalize* – наследник *KBabel*.

Теперь, наконец, приступим к самому переводу. Редактирование с этой целью строк, собранных в каталогах **po**, можно выполнить в любом текстовом редакторе, но KDE, по традиции, предоставляет нам специальный инструмент. В предыдущих версиях KDE данную функцию выполняла программа *KBabel*, а начиная с KDE 4.1 нужная утилита называется *Lokalize* (**рис. 1**).

Если открыть файл каталога в программе *Lokalize*, в списке слева (окно «Содержание») появятся строки, импортированные утилитой *xgettext* с переводами на другой язык, если таковые уже имеются. Новые переводы добавляются в окне справа. Интересно поведение программы *Lokalize* в случае, когда мы выбираем строку с несколькими формами для передачи множественного числа. При этом в окне переводов открывается несколько вкладок (как показано на рисунке). Их число соответствует значению nplurals в заголовке файла каталога (в нашем случае их будет три). Последовательность вкладок соответствует последовательности значений, возвращаемых выражением plural. В каждой из этих вкладок мы вводим перевод, не забывая указывать спецификатор %1 в подобающем ему месте.

После завершения следует должны скомпилировать файл каталога po в двоичный файл с расширением **mo**. Это делается с помощью утилиты *msgfmt*:

```
msgfmt messages.po -o i18ndemo.mo
```

Ключ -o позволяет указать имя результирующего файла, которое по умолчанию должно совпадать с именем собираемой программы. Нам осталось скопировать двоичный файл каталога в директорию, в которой приложения KDE ищут свои ресурсы локализации (программа может загрузить их только из специальной директории и никак иначе – это относится, в том числе, и к текущему рабочему каталогу). Система предоставляет две директории для размещения ресурсов локализации – глобальную, для приложений, установленных в системе, и локальную, для отдельных пользователей. Запись файлов в локальную директорию не требует прав root. По умолчанию, глобальным каталогом является общая директория ресурсов локализации **/usr/share/locale/** (в отличие от KDE 3.x, в которой использовалась собственная директория). Локальная директория расположена в **~/.kde4/share/locale/**. Файлы переводов сообщений на русский язык необходимо размещать в поддиректории **ru/LC_MESSAGES/** одной из указанных директорий. Теперь наша тестовая программа дружит с падежами (**рис. 2**).

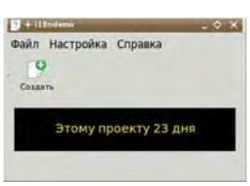


Рис. 2. Результат локализации налицо.

Контексты

В предыдущих версиях *Qt*/KDE уже были реализованы средства, призванные решить еще одну проблему перевода, связанную с многозначностью слов английского языка. Например, английское слово build может быть существительным «сборка» и глаголом «собрать». Чтобы локализаторы могли предоставить адекватный перевод одного и того же слова, используемого в разных значениях, в KDE 4 существует система контекстов. Контекст задается с помощью функции i18nc(), принимающей два аргумента: строку контекста и фразу, подлежащую переводу, например: i18nc("noun", "Build"), i18nc("verb", "Build").

Если одна и та же переводимая строка используется в разных контекстах, каждый из них вносится в файл **messages.po**. Разумеется, программа *Lokalize* позволяет ввести для каждого контекста свой вариант перевода.

Чтобы извлечь из исходных текстов строки, помеченные для перевода с помощью функции i18nc(), нужно использовать специальный синтаксис ключа -k утилиты *xgettext*. Там, где при использовании i18n() мы написали бы -ki18n, в случае с i18nc() нужно писать -ki18nc:1c,2 (иначе говоря, мы указываем, что первым аргументов функции i18nc() является идентификатор контекста, а вторым – строка, предназначенная для перевода). Как вы уже, наверное, поняли, утилита *xgettext* ничего не знает о семантике функций i18n*, а при их обработке руководствуется подсказками, которые мы передаем в ключе -k. Для обработки строк с указанием контекста и несколькими численными формами служит функция i18ncp(), которую можно рассматривать как гибрид i18nc() и i18np(). Ее первый аргумент – контекст, далее следуют строки для единственного и множественного числа и целочисленная переменная. Для извлечения строк, помеченных функцией i18ncp(), утилите *xgettext* должен быть передан ключ -ki18ncp:1c,2,3.

KLocale: даты, числа, деньги

Множество полезных функций локализации сосредоточено в классе KLocale. Его объект, управляющий локалью, создается в программе KDE автоматически. Для того чтобы получить указатель на глобальный объект класса Klocale, воспользуемся вызовом KGlobal::locale():

```
KLocale * aLocale = KGlobal::locale();
```

По умолчанию программа использует локаль, выбранную в настройках системы. Для указания локали явным образом можно использовать метод setLanguage(). В KDE 3.x ему передавалась одна строка с именем локали. В KDE 4.x мы можем передать методу список QStringList с именами нескольких локалей, из которых система выберет первую, для которой сможет найти ресурсы локализации:

```
aLocale->setLanguage(QStringList("ru"));
```

Теперь мы можем воспользоваться всеми удобствами класса KLocale. Если нам нужно вывести данные о денежной сумме в формате, соответствующем выбранной локали, к нашим услугам метод formatMoney(). Например, вызов

```
aLocale->formatMoney("10000.67");
```

вернет строку “10.000,67 руб.”. Подобным же образом работает метод formatNumber().

Для форматирования значений даты и времени нам предоставлены методы formatDate(), formatDateTime(), formatTime(). Их первым параметром должно быть значение типа QDate, QDateTime (или KDateTime) и QTime соответственно. Вторым аргументом у функций formatDate() и formatDateTime() является одна из констант типа DateFormat: ShortDate – «короткая» дата (с численным обозначением месяца), LongDate – «длинная» дата (с буквенным обозначением месяца и указанием дня недели), FancyShortDate или FancyLongDate – аналоги ShortDate и LongDate, для которых недавние (в пределах недели) даты заменяются словами «сегодня», «вчера», «понедельник» и т.д.

Помимо прочего, с помощью глобального объекта KLocale мы можем управлять загрузкой ресурсов интернационализации. Статический метод setMainCatalogue() позволяет задать имя каталога сообщений, используемого по умолчанию. Его нужно вызывать в самом начале функции main(), до того как будет создан глобальный объект KLocale. Обычно программа ищет перевод своего интерфейса в файле, название которого совпадает с ее собственным именем. С помощью метода setMainCatalogue() мы можем изменить это поведение. Так можно поступить, например, в том случае, когда фразы, требующие перевода в вашей программе, образуют подмножество строк другого, широко распространенного приложения, скажем, *Konqueror*. В этом случае, используя каталог сообщений программы *Konqueror*, вы «бесплатно» получаете перевод интерфейса вашего приложения на все поддерживаемые языки. Нестатический метод insertCatalog() класса KLocale позволяет добавить еще один элемент в список каталогов, в которых программа ищет переводы строковых ресурсов. При вызове методов загрузки каталогов имена файлов следует указывать без расширения, например:

```
KLocale::setMainCatalogue("konqueror");
```

Обновление правок

Предположим, что вы обновляете версию программы, для которой когда-то был выполнен перевод на другие языки (и, возможно, не только вами). Вы внесли некоторые изменения в интерфейс программы, но значительная часть строк осталась неизменной. У переводчиков программы хранятся файлы PO с прежним переводом и они, естественно, не хотят переводить заново то, что уже было переведено. Как перенести старый перевод в новый файл PO? Алгоритм решения таков: сгенерировать новый файл PO при помощи *xgettext*; объединить новый файл PO и старый файл с переводом с помощью утилиты *msgmerge*; перевести непереведенные строки. В простейшем случае вызов msgmerge выглядит так:

```
msgmerge старый_файл новый_файл
```

где старый_файл – файл PO с уже выполненным переводом, а новый_файл – файл PO, сгенерированный *xgettext*. По умолчанию утилита выводит результат своей работы в стандартный поток вывода, но ничто не мешает нам перенаправить его в файл. Схематично механизм слияния можно описать так: для каждой строки нового файла *msgmerge* ищет перевод в старом файле. Если перевод найден, он добавляется в результирующий текст. Таким образом, если в ходе обновления программы из ее интерфейса была удалена какая-либо строка, она уже не попадет в файл, полученный в результате слияния.

Объединять файлы каталогов можно и с помощью *Lokalize*. Загрузите основной файл каталога в *Lokalize* и выберите команду меню Синхронизация > Открыть файл для синхронизации/объединения. В появившемся окне нужно указать файл каталога для объединения с главным. В нижней части окна *Lokalize* откроется панель Синхронизация. На ней будут отображаться различия в переводах между вспомогательным и основным файлом (для этого в окне Содержание необходимо выделить строку переводимого текста, для которой различаются переводы). С помощью команд Перенести из источника слияния и Перенести все новые переводы можно добавить переводы из вспомогательного файла в основной.

Перевод из файлов ui

А как быть с теми строками, которые были заданы в файле описания интерфейса (ui)? Утилита *xgettext* не справляется с их извлечением, но нам на помощь приходит утилита *extractrc*. Эта программа извлекает помеченные для перевода строки из файлов UI и RC (имеются в виду файлы XMLGUI, а не скрипты *runcommand*) и генерирует файл **cpp**, в котором эти строки оборачиваются функциями i18n(). Утилита всегда вываливает сгенерированный текст в стандартный поток вывода, но это можно изменить перенаправлением.

Я не стал переводить интерфейс диалогового окна настроек программы *i18ndemo* – оставляю это вам в качестве домашнего задания (подсказка: для объединения существующего перевода и перевода строк интерфейса диалогового окна воспользуйтесь *msgmerge* или *Lokalize*). LXF

Через месяц Мы завершим обзор технологий KDE4 управлением устройств через Solid.

Графические

ЧАСТЬ 3 Работа в R ориентирована на консоль, но и мышь иногда тоже хочется чем-нибудь занять. **Антон Коробейников** и **Евгений Балдин** изучают графических помощников в тяжёлом деле анализа данных.

Наши эксперты

Антон Коробейников
Молодой преподаватель, читающий спецкурсы по применению R для решения различных статистических задач студентам кафедры статистического моделирования матмеха СПбГУ.

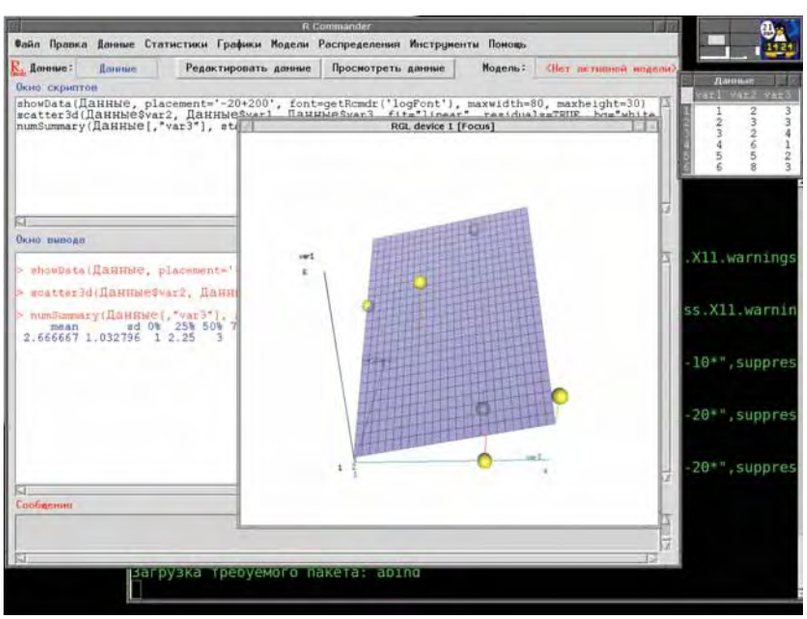
Евгений Балдин
Физик, преподаватель, научный сотрудник ИЯФ им. Будкера, профессионально занимается обработкой экспериментальных данных в области физики высоких энергий. Давний сторонник свободного ПО, впечатлённый мощью R.

Прежде чем начать повествование, хотелось бы сказать пару слов в пользу консольного интерфейса. Анализ данных – это творческий процесс, и ничто не продвигает его лучше, чем неспешный ввод с клавиатуры. Естественно, этому занятию должен предшествовать интервал времени, целиком и полностью посвящённый чтению документации, книг и статей по тематике проблемы. Менюшки и кнопочки отвлекают, создавая иллюзию простоты творческого процесса, требуя нажать их немедленно и посмотреть, что получится. Как правило, не получается ничего, т.е. всё равно приходится брать в руки книгу и думать.

Консольный интерфейс в R идеален. Он предоставляет пользователю историю команд и дополнение по Tab, сохраняет информацию и объекты между сессиями (если пользователь этого захочет, естественно). Нужно поработать на удалённом компьютере? С консолью нет никаких проблем. А если воспользоваться программой *screen*, то можно не бояться «разорванных» сессий и случайно закрытых терминалов.

Вы всё ещё хотите использовать GUI при анализе данных? Ну что же, дело ваше, но помните, что мы вас предупреждали.

› Так выглядит R Commander.



R Commander

R Commander или *Rcmdr* – кросс-платформенный графический интерфейс к R, написанный на Tcl/Tk. Его домашнюю страницу можно найти по адресу **http://socserv.mcmaster.ca/jfox/Misc/Rcmdr/**.

Автор *R Commander* Джон Фокс [John Fox] признаётся, что в случае программ для статистического анализа он не является фанатом интерфейса, состоящего из меню и диалогов. С его точки зрения, *R Commander* полезен в основном для образовательных целей при введении в R, а также в отдельных, очень редких случаях – для быстрого анализа. Одной из основополагающих целей, преследуемых при создания интерфейса *R Commander*, был мягкий перевод пользователя в консоль, где можно автоматизировать свои действия более глобально.

Пакет распространяется под лицензией GNU/GPL, и поэтому доступен во всех стандартных дистрибутивах Linux. Например, для установки в Ubuntu 8.04 достаточно выполнить команду:

```
% sudo aptitude install r-cran-rcmdr
```

Пакет присутствует также и на CRAN, поэтому его установку можно произвести и силами R:

```
> install.packages("Rcmdr", dependencies=TRUE)
```

Обратите внимание на значение опции dependencies: *R Commander* зависит от довольно большого числа пакетов. Кроме стандартных библиотек *r-base*, он предполагает, что в системе уже установлены пакеты *abind*, *aplpack*, *car*, *effects*, *lmtest*, *multcomp*, *relimp*, *rgl* и (только для Windows) *RODBC*, поэтому до инсталляции самой программы необходимо убедиться, что все они действительно присутствуют. Установка *R Commander* с помощью **aptitude/apt-get**, как и

Месяц назад Мы искали взаимосвязи между случайными величинами.

интерфейсы

добавление dependencies=TRUE, позволяет не беспокоиться об этих проблемах. Все необходимые пакеты будут скачаны и установлены в нужном порядке.

После установки, для запуска *R Commander* в сессии R следует выполнить команду:

```
> library(Rcmdr)
```

При запуске *R Commander* открывается снабжённое довольно «развесистым» меню окно, разделённое на Окно Скриптов [Script Window], Окно Вывода [Output Window], а также информационное окно Сообщения [Messages]. Многие действия в *R Commander* можно выполнять через меню, которое достаточно легко настраивается, например, с помощью редактирования текстового файла **Rcmdr-menus.txt**. Также можно вводить команды в Окне Скриптов, то есть консоль никуда не делась. При создании графиков они появляются в отдельных окнах, как и в случае R.

R Commander имеет русскую локализацию, которая активизируется автоматически, если выставлена русская локаль:

```
% locale
LANG=ru_RU.UTF-8
```

Основная проблема русской локализации – это кириллические шрифты, которые выбраны по умолчанию. Поэтому, если вы предпочитаете локализованный интерфейс, то перед загрузкой *R Commander* ему, с помощью команды options, следует передать примерно следующее:

```
> options(Rcmdr=list(
+  default.font="-rfx-fixed-medium-r-normal-*-20*",
+  suppress.X11.warnings=TRUE))
```

Растровый шрифт семейства rfx от Дмитрия Болховитянова (20 в конце строки – это размер по умолчанию), установленный в качестве default.font, идёт в составе пакета *xfonts-bolkhov*. Вторая опция, suppress.X11.warnings, подавляет надоедливые сообщения при создании новых графических окон.

Выйти из *R Commander* можно через меню Файл > Выйти, при этом можно одновременно закрыть и сессию R. Если же сессия R осталась открытой, то повторный запуск *R Commander* выполняется с помощью команды:

```
> Commander()
```

Для вводного ознакомления с возможностями *R Commander* следует прочитать текст **Getting-Started-with-the-Rcmdr.pdf**, получить доступ к которому можно через меню Помощь > Введение в R Commander.

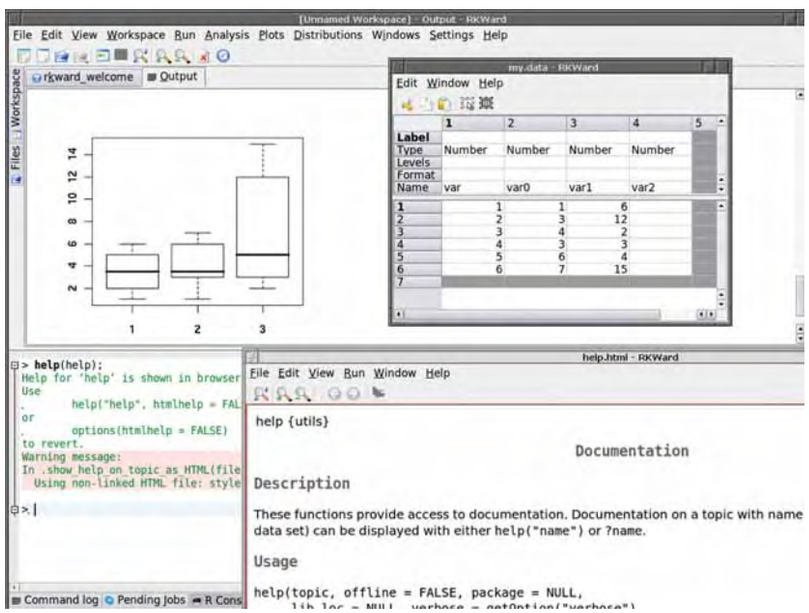
Если по какой-то причине было принято решение анализировать данные исключительно с помощью *R Commander*, то можно сделать так, чтобы при запуске R эта графическая оболочка загружалась автоматически. Для этого в файл пользовательских настроек **~/.Rprofile** достаточно добавить следующие строки:

```
old <- getOption("defaultPackages");
options(defaultPackages = c(old, "Rcmdr"));
```

Глобальная переменная defaultPackages содержит информацию о модулях, автоматически загружаемых при старте R.

RKWard

RKWard (rkward.sourceforge.net) – это довольно удобный KDE3/4-интерфейс к языку программирования R. Разработчики *RKWard* старались совместить мощь R с простотой использования, подобной предоставляемой коммерческими статистическими пакетами (такими как *SPSS* или *Statistica*).



› **RKWard – R и KDE.**

Стоит отметить, что это им вполне удалось. Для начинающих пользователей *RKWard* предоставляет широкие возможности по выбору многих стандартных процедур статистического анализа по принципу «выдели и щелкни»: достаточно активировать соответствующий пункт меню.

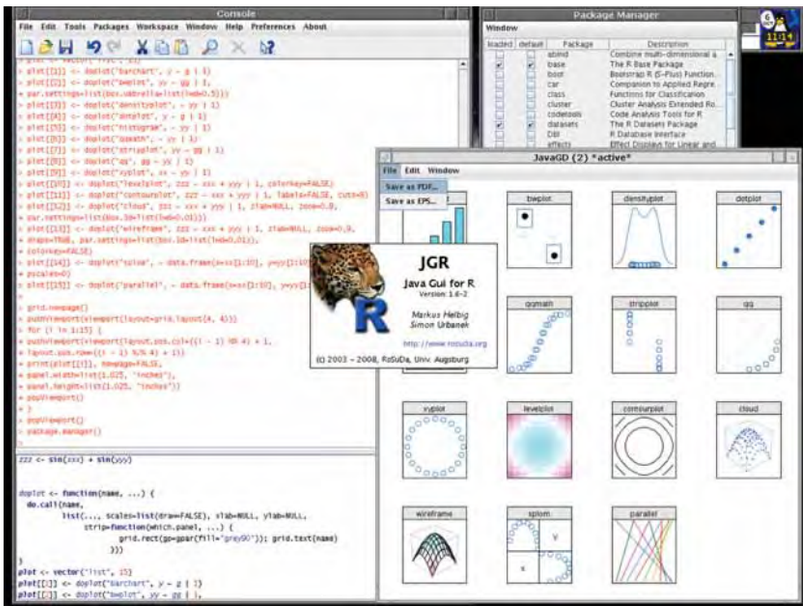
Для продвинутого пользователя *RKWard* предлагает достаточно удобный редактор кода с подсветкой, автоматической расстановкой отступов, автодополнением – теми вещами, без которых в настоящее время не обходится ни одна среда программирования. Привычная консоль R также наличествует, она доступна в любое время на вкладке R Console.

Авторы *RKWard* взяли курс на как можно более полную интеграцию функций R в графическую среду: присутствует браузер текущего окружения (environment), редактор данных. Есть менеджер пакетов, умеющий не только устанавливать их, но и следить за обновлениями; обеспечивается прозрачная интеграция со справочной системой. Кроме того, *RKWard* умеет «перехватывать» создание графических окон и добавлять к ним очень удобные функции типа сохранения содержимого в файл одного из стандартных форматов, поддерживаемых R (PDF, EPS, JPEG, PNG).

Интерфейс *RKWard* чрезвычайно гибок: пользователь может расширять его за счёт написания собственных модулей (кстати, все встроенные средства анализа, доступные сразу же после запуска из меню – это такие же модули, но только созданные авторами *RKWard*).

JGR

Предпочитаете, чтобы интерфейс отрисовывался средствами Java? Тогда *JGR* (Java GUI для R) – это то, что вы искали. *JGR* (произносится как «ягуар») был впервые представлен публике в 2004 году, но развивается и поддерживается до сих пор. Пакет *JGR* распростра-



JGR – интерфейс со вкусом Java.

няется под лицензией GPL2, и его домашняя страница доступна по адресу http://jgr.markushelbig.org/JGR.html.

JGR создавался для Mac «со всеми вытекающими», но под Linux он тоже работает. Для использования *JGR* необходимо инсталлировать Sun Java Development Kit (JDK).

```
> apt-get install sun-java6-jdk
> sudo update-java-alternatives -s java-6-sun
> sudo  R CMD javareconf
```

После установки и настройки Java-окружения следует запустить сессию R и выполнить следующие действия:

```
> install.packages('JGR')
> library(JGR)
> JGR()
Starting JGR run script. This can be done from the shell as well,
just run /home/user/R/i486/2.6/JGR/scripts/run
```

Готово – *JGR* запущен. Как гласит сообщение, то же самое можно проделать из консоли, набрав путь до специального скрипта. В приведённом выше примере это **/home/user/R/i486/2.6/JGR/scripts/run**.

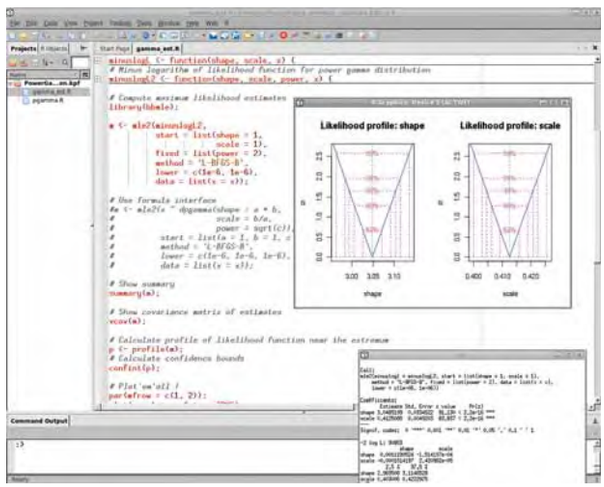
В *JGR* есть встроенный текстовый редактор, который, конечно, не является полноценной заменой *Emacs/VIM*, но в нём есть подсветка синтаксиса и Tab-завершение команд. В *JGR* встроена гипертекстовая помощь, простенькая электронная таблица, добавлена возможность управлять объектами, в том числе и с помощью мыши, а также есть графический интерфейс для установки и загрузки R-пакетов. С учётом того, что этот GUI может работать везде, где есть Java, на него стоит обратить внимание.

SciViews-K

В мире универсальных редакторов и IDE достаточно известна платформа разработки Komodo от ActiveState. Она зарекомендовала себя как удобная среда для языков с динамической типизацией: PHP, Python, Perl, Ruby. *SciViews-K* (www.sciviews.org/SciViews-K) – молодой и динамично развивающийся проект (первый релиз состоялся в июне 2008 года), добавляющий поддержку R к свободному (MPL/GLP/LGPL) редактору *Komodo Edit*.

На настоящий момент возможности *SciViews-K* довольно скромны: более чем обычный редактор кода с подсветкой, интеграция с командной строкой R и системой справки. Кроме того, доступен просмотр текущего окружения и редактор наборов данных.

Поскольку *SciViews-K* является расширением для редактора *Komodo Edit*, то его установка производится в два приема: сперва необходимо инсталлировать сам *Komodo Edit*, а потом уже добавить *SciViews-K*. Кроме того, при первом запуске R из-под *SciViews-K* будет установлен пакет *SciViews* вместе с зависимостями.



***SciViews-K* – расширение для Komodo Edit.**

Rattle

Rattle (rattle.togaware.com) – сокращение, обозначающее «the R Analytical Tool To Learn Easily» [Легкая в освоении среда анализа R]. Программа активно развивается и, по уверениям разработчиков, предназначена для интеллектуального анализа данных (data mining), иными словами, для выявления скрытых закономерностей или взаимосвязей между переменными в больших массивах необработанных данных. *Rattle* – это среда для «разглядывания» данных человеком, и программа всячески помогает ему в этом деле. Альтернативное название пакета: «A GNOME Data Miner Built on R» намекает, что этот пакет интегрирован в графическую среду GNOME.

Для установки *Rattle* необходимо наличие пакетов *r-base-core*, *ggobi* (визуализация данных) и *libglade2-dev*:

```
> apt-get install r-base-core ggobi libglade2-dev
```

После этого в консоли R следует выполнить команду:

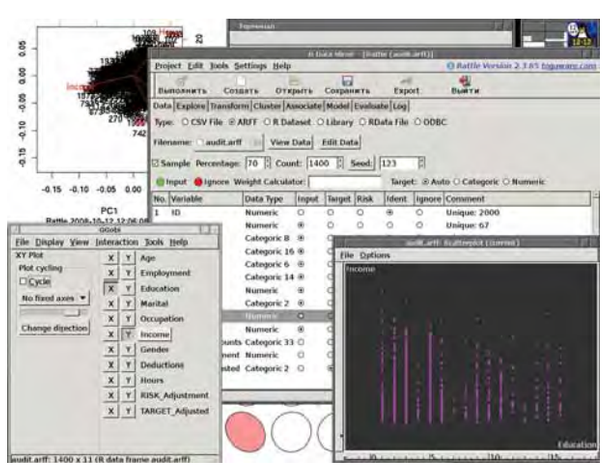
```
> install.packages("rattle", dependencies=TRUE)
```

и откинуться на спинку кресла. Из-за большого количества зависимостей установка занимает много времени.

Запуск GUI производится как обычно:

```
> library(rattle)
> rattle()
```

Простой интерфейс позволяет использовать *Rattle* для обучения основам анализа.



***Rattle* и анализ данных в нём.**

Второй эшелон

PMG

Нужен графический интерфейс, но нет особого желания разбираться в «навороченных» GUI? Тогда, возможно, имеет смысл обратить внимание на пакет *PMG* – Poor Man's GUI (**wiener.math.csi.cuny.edu/pmg**).

Хотя у пакета есть множество зависимостей, установка проста и проводится силами R. *PMG* является кроссплатформенным ПО: для отрисовки графического интерфейса применяется библиотека *GTK* в форме ***Rgtk2***. Последний, в свою очередь, можно использовать для создания собственного GUI.

```
> install.packages("pmg", dep=TRUE)
> library(pmg)
```

Этому графическому интерфейсу определённо есть куда развиваться.

RPMG

Что делать, если нет необходимости в сложной переливающейся графике, а нужны пара простейших интерактивных кнопочек на картинке и минимальное число зависимостей от других пакетов? В этом случае поможет *RPMG* – GUI для действительно бедных (Really Poor Man's GUI): с его помощью легко можно организовать интерактивное графическое R-окно для личного пользования. Для установки пакета и демонстрации его возможностей следует выполнить следующие команды:

```
> install.packages("RPMG", dep=TRUE)
> library(RPMG)
> demo(RPMG)
```

Пакет не имеет домашней страницы, но его всегда можно скачать со CRAN: **http://probability.ca/cran/web/packages/RPMG/index.html**.

RWeb

Допустим, вы не хотите давать в руки своим студентам ничего опаснее web-браузера. Странное желание, но и оно не помеха для преподавания статистики с использованием R. Простой набор Perl-скриптов *Rweb* (R Web Based Statistical Analysis), доступный на **www.math.montana.edu/Rweb**, можно использовать как отправную точку для реализации своего интерактивного web-проекта. На сайте проекта есть его рабочая демонстрация: команды R можно набирать прямо в окне браузера, здесь же появляются результаты. Проект давно не обновлялся, но его код вполне обозрим и его можно легко подстроить под свои нужды.

Gnumeric

Gnumeric (**http://www.gnome.org/projects/gnumeric**) – это популярный табличный процессор с открытым исходным кодом, иными словами – электронная таблица, со всеми преимуществами и недостатками данного типа программ. *Gnumeric* поддерживает систему подключаемых модулей, позволяющая расширять функциональность. В 2003 году сотрудник Bell Labs Дункан Лэнг [Duncan Temple Lang] написал дополнение, предоставляющее возможность вызвать из *Gnumeric* любую функцию R. Этот модуль был незатейливо назван *RGnumeric*, и его до сих пор можно найти по адресу **www.omegahat.org/RGnumeric**.

К сожалению, пакет для интеграции R с *Gnumeric* представляет из себя скорее демонстрацию возможностей, нежели законченный продукт. В рамках Google Summer of Code 2008 была попытка возродить заложенные в него идеи на новом уровне (**http://www.r-project.org/SoC08/ideas.html**), но похоже, проект всё ещё ждёт своего автора. В любом случае, исходники доступны и, хотя просто сама их сборка представляет из себя некоторое приключение, истинные энтузиасты не ищут лёгких путей.

Emacs

Что это мы всё о GUI да о GUI? Пора поговорить о *Emacs*. Да-да, про тот самый редактор, в который встроена программа управления кофеваркой. Говорят, она работает, причём очень неплохо. А что же R?

ESS (**ess.r-project.org**) – это специализированная интерактивная среда или мода для GNU Emacs/XEmacs. *ESS* является сокращением от фразы «Emacs Speaks Statistics», которую можно перевести как «Emacs говорит на языке статистики». Пакет *ESS* поддерживает не только систему статистического анализа R, но и другие диалекты языка S (S 3/4, S-PLUS 3/4/5/6/7), а так же SAS, XLispStat, Stata и BUGS. При установке в пакета ESS, например, так:

```
> sudo aptitude install ess
```

и добавлении строчки для инициализации *ESS* в **~/.emacs**

```
(require 'ess-site)
```

редактору *Emacs* становятся доступны две дополнительные моды:

- Мода ESS, которая предназначена для редактирования исходных файлов с использованием команд R (включая дополнительную поддержку для редактирования Rnw-файлов), и
- Мода iESS – замена обычной R-консоли.

Интерпретатор R запускается, как это ни странно, с помощью команды R: Alt+x, а затем R и Enter. При старте спрашивается о местоположении рабочей директории для открываемой сессии R, а затем отображается стандартное R-приглашение. Можно параллельно запустить насколько сессий, а также задействовать удаленный компьютер (команда ess-remote). Работа в этой моде почти ничем не отличается от консоли, за исключением наличия стандартных возможностей редактирования текста в *Emacs* и дополнительного вспомогательного меню iESS. Из него можно получить доступ к таким интересным возможностям, как, например, редактирование объектов R (**^c^d**). *Emacs* – это прежде всего текстовый редактор, поэтому подобные особенности делают анализ данных комфортнее.

ESS-мода инициализируется автоматически при редактировании файлов с расширением R. *Emacs* предоставляет пользователю подсветку синтаксиса, выравнивание исходного кода по Tab, выполнение отдельных строк, фрагментов и всего файла в интерпретаторе R. Дополнительное меню *ESS* позволяет получить доступ ко всем этим возможностям. Редактирование RNW-файлов (смесь LaTeX- и R-команд для быстрого составления отчётов с помощью пакета *Sweave* – см. LXF102) чуть более мудрёное: мода *ESS* активируется только когда курсор оказывается внутри фрагментов с R-командами.

Если у вас есть навыки работы в *Emacs*, то *ESS* – это верный выбор.

И ещё...

R широко известен (хоть и в довольно узких кругах), поэтому почти все уважающие себя текстовые редакторы или среды разработки поддерживают его в той или иной степени. Кроме уже упомянутого *Emacs*, к ним относится и *Vim*, и *jEdit*, и *Bluefish*, и *SciTE*. Даже *Eclipse* имеет соответствующий модуль (**http://www.walware.de/goto/statet**). Сегодня мы говорили только о свободных пакетах, но, потрудившись, можно будет найти поддержку и в коммерческом секторе. LXF

Через месяц Мы займемся интеллектуальным анализом данных, также известным как Data Mining.

gLite Введение в GRID-технологии на основе Scientific Linux.

«Грид»: что,

ЧАСТЬ 1 Слово «грид» нынче в моде, но лишь немногие в курсе, что оно означает. И уж совсем немногие знают, что оно имеет отношение к Linux. **Андрей Е. Шевель** поможет вам попасть в число этих немногих.

Наш эксперт

Андрей Е. Шевель
Зав. отделом вычислительных систем в Отделении физики высоких энергий Петербургского института ядерной физики Российской академии наук. Основной научный интерес – распределенные вычислительные системы.

На данном уроке мы сделаем краткий обзор простейших возможностей, особенностей и терминологии распределенных вычислительных систем с архитектурой «грид» (grid) на основе комплекта пакетов промежуточного ПО *gLite*. Он был сформирован для решения задач распределенной обработки данных в экспериментах на ускорителе элементарных частиц Large Hadron Collider (LHC) – Большой Адронный Коллайдер (БАК) в Женеве (Швейцария). Несмотря на то, что составители комплекта имели в виду в основном научную обработку данных, *gLite* представляет собой весьма общий набор инструментов для организации распределенной обработки в любой отрасли.

Немного теории

В начале 90-х несколько ученых, работавших в области вычислительной техники (среди них было несколько лидеров современного консорциума Globus), пришли к мысли, что для проведения больших вычислений в различных областях знаний (моделирование мировой экономики, климата планеты или отдельного региона, рождения звезд, ядерных взрывов) недостаточно традиционного на тот момент подхода: проведения расчетов в одном/двух местах с использованием либо больших монолитных вычислителей, либо кластеров, нередко – специально настроенных под конкретный класс задач. Так, кластер, настроенный для моделирования океана, лишь с трудом мог бы быть использован, скажем, в экономике.

Имелся целый ряд причин, которые препятствовали использованию одного и того же кластера в различных отраслях и даже разными организациями: например, применялись ПО и оборудование, хорошо подходившее для одних задач и не слишком – для других. Имели место и административные, и финансовые барьеры: различные кластеры приобретаются и поддерживаются из разных фондов. При использовании удаленных кластеров должны были решаться проблемы сетевой безопасности, причем таким образом, чтобы пользователю не приходилось постоянно вводить пароль (иначе затрудняется выполнение автоматизированных скриптов)... Перечисление можно продолжить.

С другой стороны, для больших задач, как правило, недостаточно одного кластера. Иными словами, возникает необходимость интегрировать ресурсы многих кластеров, которые, вообще говоря, распределены по странам и континентам, организациям и компаниям, зданиям и офисам. Более того, кластеры имеют различное «подчинение», различные источники финансирования, различный стиль работы администраторов. Как объединить столь непохожие стихии?

Во-первых, понадобилось разработать программное обеспечение, которое дало технические предпосылки для объединения разнородных вычислительных установок для нескольких классов задач:

- Больших вычислительных проблем, где требуется главным образом процессорное время.
- Больших задач, где требуется использовать огромный объем данных, много петабайтов.
- Больших задач, в решение которых вовлечено значительное число специалистов, проживающих в разных городах, странах и континентах.

Естественно, что имеются проблемы, которые совмещают все перечисленные требования.

Процесс осознания полезности интегрирования распределенных компьютерных ресурсов начался в явном виде с выхода сборника статей нескольких десятков авторов, объединенных в книгу «The Grid: blueprint for a new computing infrastructure» [Сеть: генеральный план новой компьютерной инфраструктуры], которая была опубликована под редакцией Яна Фостера [Ian Foster] и Карла Кесельмана [Carl Kesselman] в 1999 году (см. www.amazon.com/Grid-Blueprint-New-Computing-Infrastructure/dp/1558604758). В книге были последовательно описаны методы и средства вычислений, используемые в различных отраслях научной и инженерной деятельности: в физике, химии, медицине, геологии, биологии, космических исследованиях и т.д. Даже беглый взгляд обнаруживал, что в методах самых разных областей знания имеется масса общего. Были выделены основные классы вычислений.

- Распределенные вычисления.
- Высокопроизводительные вычисления.
- Вычисления по требованию, т.е. такие, которые не требуют постоянного доступа к вычислительным ресурсам, а лишь по мере необходимости.
- Обработка больших объемов данных.
- Коллаборационные вычисления (где участвует целая армия специалистов, рассеянных по всему свету).

Естественно, сразу возникли мысли о том, как организовать компьютерную инфраструктуру таким образом, чтобы можно было интегрировать необходимые ресурсы почти автоматически. Иными словами, использовать их так, как мы сейчас используем электрическую сеть, т.е. создать некоторый вычислительный grid (грид). Здесь уместно заметить, что этот термин есть литературная метафора, а не строго определенное понятие.

Для реализации такого амбициозного подхода понадобилось решение нескольких задач:

откуда и зачем?

» Безопасность, которая включает аутентификацию пользователя (имя пользователя, его идентификатор, полномочия, принадлежность к конкретной группе пользователей – виртуальной организации).

» Универсальное программное окружение, которое позволяло бы определить свойства кластера и его узлов, не выполняя вход в систему – иными словами, специальный сервер, который бы автоматически собирал с кластеров информацию об их состоянии (доступен, недоступен) и архитектурных особенностях (число доступных узлов, состав узлов, прочее). Клиент на вашем компьютере может связаться со специальным сервером, «ресурсным брокером», который подберет кластер, наиболее подходящий для вашего задания. На первый взгляд, задача может показаться тривиальной, но если число кластеров составляет многие десятки, сотни или тысячи, то ответ уже не столь очевиден.

» Мониторинг ресурсов различных кластеров, линий связи в Интернете с тем, чтобы иметь представление о загруженности тех или иных ресурсов.

» Передача данных от кластера к кластеру. Средства обеспечения надежной передачи больших объемов информации за минимальное время и в каком-то смысле оптимальным образом. Создание копий (реплик) данных на нескольких кластерах.

» Наконец, администрирование больших групп пользователей – виртуальных организаций.

Инициатива стартовала примерно в 2000 году. Почти в одно время начались работы над несколькими проектами в разных организациях: исследовательских центрах, университетах. Среди первых, естественно, были те группы и организации, которые уже столкнулись с задачами, упомянутыми выше:

» интернациональные группы ученых (коллаборации) в области исследования космического пространства и исследования земли с космических аппаратов;

» колаборации исследователей в области биологии и медицины;

» коллаборации исследователей по изучению свойств материи на ускорителях заряженных частиц (экспериментальная ядерная физика и физика высоких энергий).

Все разработки такого типа (грид) выполнялись совместно, большим числом специалистов из самых разных отраслей знаний. Было создано большое число сайтов, посвященных грид-разработкам. Среди основных можно отметить **www.globus.org** – исторически первый сайт разработчиков грид-продуктов; **www.ggf.org** – свободное международное объединение разработчиков для обсуждения и продвижения перспективных средств распределенных вычислений. Среди русскоязычных ресурсов сайтов упомянем **www.egee-rdig.ru** и **www.gridclub.ru**. Читатель запросто сможет найти сотни других привлекательных сайтов на данную тему.

В течение нескольких лет в рамках ряда проектов были созданы десятки решений, которые стали называть middleware (промежуточное ПО, т.е. такие программы/библиотеки, которые служат интерфейсами между различными программными слоями). Среди крупнейших можно упомянуть проекты: Open Science Grid (OSG) – **opensciencegrid.org**, Enabling Grids for E-sciencE (EGEE) – **public.eu-egee.org/intro**, NorduGrid – **www.nordugrid.org**.

Пакеты-middleware

Почти все разработки в области грид-архитектур являются открытыми, а созданное ПО – свободно распространяемым, поэтому Linux, очевидно, является для них естественной платформой. Для этих целей мировое научное сообщество организовало специальный дистрибутив – Scientific Linux (**www.scientificlinux.org**). Основой для него на сегодняшний день служат исходные тексты Red Hat Enterprise Linux. В настоящее время Scientific Linux используется на многих тысячах компьютеров и сотнях кластеров в различных уголках планеты. Таким образом, надежность этой системы наравне с приемлемой функциональностью не вызывает сомнений. Российское зеркало сайта Scientific Linux вместе с переводом описаний доступно по адресу **www.scientificlinux.ru**.

Десятки различных разработок в области грид-систем вылились в естественный параллелизм в каких-то аспектах. Поэтому, когда конечный пользователь пытается применять относительно разрозненные программные пакеты, могут возникать нестыковки, затрудняющие использование модели «грид» в конкретном окружении.

В связи с этими обстоятельствами, крупные проекты составили свои комплекты пакетов промежуточного ПО с таким расчетом, чтобы конечный потребитель смог использовать модель вычислений «грид» не только в многолюдных коллаборациях, но и в относительно небольших коллективах. К наиболее известным комплектам промежуточного ПО относятся:

» *Advanced Resource Connector* (**www.nordugrid.org/middleware**) – продвинутый исследовательский коннектор из проекта NorduGrid;

» *Virtual Data Toolkit* (**vdt.cs.wisc.edu//index.html**) – виртуальный инструментарий для управления данными из проекта **OSG**;

» *gLite* (**www.glite.org**) – часть проекта **EGEE**.

Последний из них мы сейчас и рассмотрим более подробно.

WLCG

За данной аббревиатурой скрывается словосочетание Worldwide LHC Computing Grid – Всемирный вычислительный грид для Большого Адронного Коллайдера. Все пользователи WLCG/EGEE объединены в ряд групп – виртуальных организаций (virtual organization, VO). Каждая VO (и, следовательно, каждый член виртуальной организации) имеет доступ к некоторому набору компьютерных ресурсов, распределенных по планете. Любой пользователь, который планирует работать в рамках конкретной виртуальной организации, должен зарегистрироваться, т.е. внести свои персональные данные. В рамках регистрационной процедуры пользователь получает специальный электронный личный сертификат, который выдается центром сертификации – Certification Authority (CA). Сертификат представляет собой два небольших файла с закодированной информацией о пользователе, организации, выдавшей сертификат, сроке его действия и другими служебными данными (LXF93). Один из файлов содержит открытый ключ пользователя, второй – секрет-

ный (закрытый) ключ. Последний защищен паролем, который знает лишь его владелец.

После успешной регистрации пользователь может получить доступ к ресурсам WLCG/EGEE с использованием выданного ему ранее личного сертификата. Доступ к ресурсам VO осуществляется посредством инфраструктуры грид-безопасности – Grid Security Infrastructure (GSI). GSI позволяет опознать пользователя по его сертификату, обеспечив защищенный обмен данными через открытую компьютерную сеть (локальную или Интернет). Технически, GSI базируется на использовании Public Key Infrastructure (PKI) – инфраструктуры шифрования с открытым ключом с использованием стандарта X509.

Секретный ключ пользователя применяется для генерации другого, временного (действующего обычно несколько часов или десятков часов) сертификата, так называемого прокси (proxy certificate). Этот прокси используется в гриде для опознания пользователя на различных сервисах, а также для делегирования полномочий. Таким образом, единожды введя пароль для секретного ключа при создании прокси, пользователь может не делать этого в дальнейшем, поскольку инфраструктура GSI будет применять его прокси каждый раз, когда потребуется аутентификация на каком-то из сервисов грида. Этот процесс может происходить двумя путями:

» Специальное имя пользователя из прокси сравнивается со списком имен на хосте в файле *grid-mapfile*. Если специальное имя пользователя обнаруживается, то там же будет указано, какую учетную запись следует использовать на данном хосте.

» Второй метод основывается на использовании факта принадлежности к конкретной VO, механизме Virtual Organization Membership Service (VOMS) и сопутствующих служб.

Создание временного прокси выполняется командой

```
woms-proxy-init
```

Эта утилита попросит ввести пароль (вернее, парольную фразу), который был использован при получении сертификата. После инициализации прокси можно пробовать запускать задания.

Запуск заданий

Задание(я) формируется пользователем с помощью специального языка описания заданий – Job Description Language (JDL). Например

```
Executable  = "/bin/date";
StdOutput   = "std.out";
StdError    = "std.err";
```

Естественно, настоящие задания выглядят сложнее. Запуск выполняется командой обращения к Workload Management System (WMS) – специальной системе управления загрузкой заданий. Более сложные задания могут содержать ссылки на грид-файлы, которые рассматриваются ниже, и рабочие области данных (группа файлов или директория), которые в большинстве грид-систем называются «песочницей» (sandbox). Имеются вводные «песочницы», которые нужны для запуска задания, и выводные «песочницы», которые сформированы выполненным заданием.

Запущенное задание в *gLite* может находиться в нескольких состояниях (см. рис. 1):

» **SUBMITTED** – задание запущено пользователем, но еще не обработано сетевым или прокси-сервером планирования запуска (WM).

» **WAITING** – задание обработано сервером прокси, но еще не обслужено сервером загрузки.

» **READY** – заданию уже назначен сайт, где оно будет выполняться, но оно пока не передано на этот сайт.

» **SCHEDULED** – задание ожидает в очереди на исполнение на сайте, где оно должно быть выполнено.

» **RUNNING** – задание выполняется.

» **DONE** – задание завершено.

» **ABORTED** – задание завершено аварийно системой планирования загрузки.

» **CANCELLED** – задание завершено досрочно по команде пользователя.

» **CLEARED** – рабочие области задания («песочница») переданы на компьютер пользователя.

Перед запуском полезно выяснить, какие, собственно, ресурсы будут доступны для выполнения такого простого задания. Это можно сделать командой

```
glite-wms-job-list-match -a JobShownAbove.jdl
```

Здесь может быть показан список сайтов (кластеров), где имеет смысл выполнять данное задание. Запустить же его можно командой

```
glite-wms-job-submit -a JobShownAbove.jdl
```

Если все пройдет нормально, вы получите в ответ строку, которая представляет собой идентификатор задания, вроде

```
https://lcg16.sinp.msu.ru:9000/Ug5Vkpwd14zONrfndJ_Iyg
```

Заметим, что это не обычный URL, а просто уникальный идентификатор задания. Используя его, можно посмотреть текущее состояние задания командой

```
glite-wms-job-status jobId
```

Аналогично, чтобы прекратить выполнение задания, применяется команда

```
glite-wms-job-cancel jobId
```

Можно запускать как неинтерактивные (пакетные), так и интерактивные задания, включающие взаимодействие с пользователем, задания с использованием MPI, организацию больших групп заданий и т.д. Кроме этого, существует возможность описания взаимозависимости заданий. Например, задание A1 должно выполняться раньше заданий A2 и A3, причем задание A3 должно запускаться на исполнение только после успешного выполнения заданий A1 и A2. Наконец, возможно использовать так называемые параметрические задания, когда необходимо запустить серию однотипных вычислений, отличающихся одним параметром.

Возникает естественный вопрос: а где же выполняются эти задания? Где хранятся данные? В терминологии проекта *gLite* для этого имеется специально организованные сайты и серверы:

» Сайт *«вычислительный элемент»* (Computational Element – CE). Обеспечивает выполнение заданий; иными словами, за термином CE скрывается вычислительный кластер. Как правило (но не обязательно), недалеко от CE имеется один или несколько так называемых *«элементов памяти»*.

» Сайт *«элемент памяти»* (Storage Element – SE). Это сайт, который обеспечивает хранение и передачу данных в больших объемах. Элемент SE может быть использован как ближайшим CE, так и многими другими, в том числе удаленными.

» Сервер *«рабочий узел»* (Worker Node – WN). Термином WN обозначается один из узлов компьютерного кластера, т.е. один из нескольких (многих) компонентов CE.

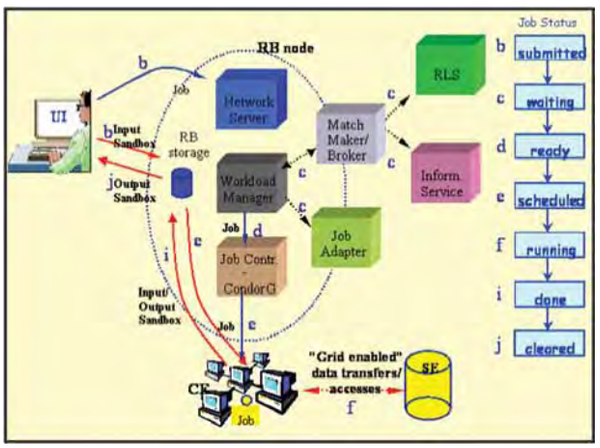
» Сервер *«пользовательский интерфейс»* (User Interface – UI). Термин UI означает любой компьютер (сервер, настольный ПК или ноутбук), где установлено клиентское программное обеспечение системы *gLite*.

Сайты (CE и SE) могут физически находиться как в одном помещении (например, в машинном зале), так и в разных городах, странах, на разных континентах или планетах.

Таким образом, задание, запущенное в приведенном выше примере, выполняется на одном из доступных CE.

На **рис. 1** представлена блок-схема выполнения задания в гриде. В колонке справа приведена последовательность состояний задания в процессе выполнения (они отмечены буквами от «b» до «j»). Слева имеется блок-схема различных элементов грид-архитектуры, которые обрабатывают задание. Слева вверху мы видим, как пользователь запускает задание с компьютера UI («пользовательский интерфейс»). Действию пользователя по запуску задания (команда glite-wms-job-submit) приписана литера a, которая не показана на рисунке. Далее UI связывается с сервером Resource Broker (RB) – брокером ресурсов грида. Чтобы найти подходящий вычислитель-

Иллюстрация взята из презентации «Программный пакет gLite 3» Д. Заборова: egee.itep.ru/gLite3.ppt.



› **Рис. 1. Блок-схема прохождения заданий.**

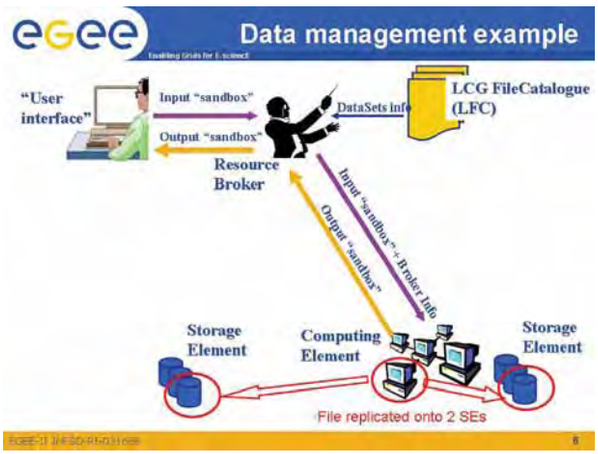


Иллюстрация взята из презентации EGEE Tutorial, 23rd APAN Meeting, Manila, Jan 22, 2007.

› **Рис. 2. Роль брокера ресурсов в *gLite*.**

ный элемент CE и, если необходимо, элемент SE, сервер RB использует другие информационные системы грида:

» **Replica Location Service** (RLS) – сервис нахождения реплик файлов (с использованием LFC);

» **Information Service** (IS) – информационный сервис, который предоставляет сведения о ресурсах (CE, SE), доступных в настоящее время (мы займемся ими в следующий раз).

» **Match-Maker-Broker** – сервис, который производит подбор подходящих элементов CE и SE для выполнения задания.

В конечном счете брокер ресурсов играет роль дирижера при запуске задания (см. **рис. 2**).

Как мы видим, UI передает на RB само задание и вводную «песочницу» – это стадия **b** (SUBMITTED) на **рис. 2**. Далее производится выбор подходящих элементов для исполнения задания – стадия **c** (WAITING). Затем выполняется передача задания на соответствующий элемент CE и другие служебные операции для постановки задания в очередь на конкретном CE – **d** (READY). После установки в очередь на конкретном кластере (элементе CE) задание находится в стадии **e** (SCHEDULED). Здесь оно обслуживается системой пакетной обработки заданий (одной из следующих: Condor, LSF, PBS {Pro|Open|Torque}, SGE). Во время выполнения задание может использовать данные из элемента SE или генерировать новые файлы и записывать их в элемент SE с использованием грид-инструментария – стадия **f** (RUNNING). Наконец, задание завершается – стадия **i** (ENDED). Финальная часть для грид – стадия **j** (CLEARED), когда выводная «песочница» пересылается в финальную точку.

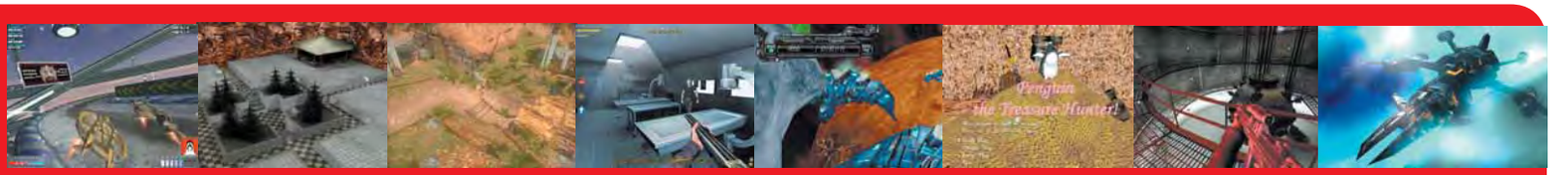
Итак, мы рассмотрели запуск и жизненный цикл простых заданий. Ответ на второй вопрос – откуда эти задания берут данные – и некоторые другие будет дан в следующий раз. LXF

» **Через месяц** Доступ к данным, передача файлов и информационный сервис *gLite*.

Технология счастья

SUNRADIO.RU

сетевое радио под ключ на базе Linux
новое будущее вашей компании
pr@sunradio.ru +7 812 955 76 70 www.sunradio.ru



GLSL Проникаем в тайны видеоускорителя с программами на шейдерах.

АНДРЕЙ ПРАХОВ
Участник нескольких игровых проектов, представитель СМИ и начальник
www.linuxmedia.ru

Игры для Linux в Сети

Несмотря на пронзительные вопли некоторых несознательных личностей, что для нашей любимой ОС нет достойных игр, разработчики не сидят на месте, а спокойно клепают «тайтл» за «тайтлом». Но обратите внимание, что сказанное относится, в основном, к онлайн-играм. То ли разработка клиента обходится относительно дешево, то ли это просто выгодно. Действительно, тенденции в современном мире таковы, что Интернет доступен почти для каждого. Человек, как известно, любит вращаться в кругу себе подобных, даже в виртуальном мире. И разработка клиента под другую ОС приносит дополнительную выгоду в виде играющих линуксоидов. Так что же нам может преподнести мир онлайн-игр? *Darkorbit*, *EVE Online*, *Outer Space*, *TimeZero*, *UT3*. Список может продолжаться и продолжаться. Так может хватит ныть и пора поиграть? Google вам в помощь...

Текстуры на заказ

ЧАСТЬ 2 Оказывается, текстуры можно создавать не только в графическом редакторе! **Андрей Прахов** умеет делать это на лету, при помощи… правильно, программ-шейдеров.

Когда-то, много лет назад, я увлекался программированием на ассемблере для микрокомпьютера ZX-Spectrum. В наше время его технические характеристики могут вызвать разве что улыбку: ОЗУ 48-1024 КБ, процессор 3,4 МГц. Однако, несмотря на крохотную память, малые тактовые частоты и полное отсутствие какого-либо ускорения графики, на нем делались потрясающие игры и демо. Существовало повальное увлечение демо-сценой, когда программист старался вместить музыку, графику, спецэффекты в ничтожное количество байт. Современные игры пожирают гигабайты места на винчестере и многие мегабайты простой и видеопамяти. Откуда такие аппетиты? Наиболее ресурсоемкие части игры – это графика и музыка. Особенно трудно приходится программистам при расчете объема текстурной памяти, необходимого для определенного момента текущей сцены. Уменьшить объем используемой видеопамяти можно, если заменить обычные текстуры процедурными, то есть вычисляемыми в нужный момент времени. В этом случае ОЗУ расходуется только на генерирующую программу. Понятное дело, это требует немалых вычислительных мощностей, если не поручить ее выполнение GPU, а точнее – его шейдерным конвейерам.

Специально для этого урока я подготовил модель дома с окружением, но вот основные текстуры для него мы заменим на процедурные программы. Итак, нам предстоит разработать шейдеры для основания дома, стен, окон и двери, т.е. создать имитации мрамора, дерева и кирпича. Давайте прикинем, сколько же мы на этом сэкономим? Если брать за стандарт текстуру формата Targa с разрешением 1024x1024, то в сумме получится... ого, примерно шесть мегабайт! За них стоит побороться.

Джек, который построил дом...

Вначале определимся с вершинным шейдером, который будет одинаковым для всех примеров. Возьмем за основу код вычисления диффузного освещения из предыдущего урока и немного переделаем его:

```
//GLSL vertex shader
void main ()
const vec4 inColor = vec4 (1.0, 0.0, 0.0, 1.0);
uniform vec3 lightPos;
```

В прошлый раз Мы узнали, что такое шейдер и немного поиграли с освещением.

```
varying vec4 outColor;
{
        vec3 position = vec3 (gl_ModelViewMatrix * gl_Vertex);
        vec3 lightvec = normalize (vec3 (lightPos) - position);
        vec3 norm = normalize (gl_NormalMatrix * gl_Normal);
        outColor = inColor * (max (dot (norm, lightvec), 0.0));
        gl_Position = gl_ModelViewProectionMatrix * gl_Vertex;
        gl_Position = ftransform();
}
```

В приведенном выше примере мы передавали координаты источника света через переменную LightPos. На этом уроке будет удобнее брать их непосредственно из самого шейдера. Раз у нас всего один источник света, то строка будет выглядеть следующим образом:

```
vec3 ll=gl_LightSource[0].position;
```

Так как основные манипуляции будут происходить во фрагментном шейдере, то логичнее было бы вычислять не полное освещение и цвет текущей вершины, а всего лишь интенсивность светового потока. В приведенном коде за это отвечает строка

```
outColor = inColor * (max (dot (norm, lightvec), 0.0));
```

Здесь происходит умножение постоянного цвета примитива (переменная InColor) на интенсивность освещения. Соответственно, удалив первый операнд, мы получим необходимый код. Заодно поменяем название выходной varying-переменной на более подходящее имя outLight. В дальнейшем останется только умножить результат работы фрагментного шейдера на outLight для получения конечного результата. Кроме того, для работы нам понадобятся координаты рассчитываемой вершины. Введем для этого новую переменную MCposition. Реализация последней задачи будет немного варьироваться для каждого конкретного шейдера, и мы обговорим ее отдельно.

Со всеми приведенными изменениями у нас получился следующий код вершинного шейдера:

```
//GLSL vertex shader (optimization for break)
void main ()
varying float outLight;
varying vec2 MCposition;
{
        vec3 ll=gl_LightSource[0].position;
        vec3 position = vec3 (gl_ModelViewMatrix * gl_Vertex);
        vec3 lightvec = normalize (vec3 (ll) – position);
        vec3 norm = normalize (gl_NormalMatrix * gl_Normal);
        outLight = max (dot (norm, lightvec), 0.0);
        gl_Position = gl_ModelViewProectionMatrix * gl_Vertex;
        MCposition = gl_Vertex.xy;
}
```

Существуют два варианта создания процедурной текстуры: использование особого алгоритма для повторяющихся элементов рисунка и работа с шумовыми функциями. Сначала мы разработаем шейдер для имитации кирпичной стены. В этом случае наиболее естественным выглядит первый вариант.

Создаваемая нами программа должна учитывать следующие особенности:

- Кирпичная кладка состоит из блоков одинакового размера, где каждый кирпич отделен друг от друга пространством другого цвета;
- Каждый следующий ряд сдвинут по отношению к предыдущему на половину длины кирпича.

Давайте определимся с необходимыми переменными. Это uniform-переменные для хранения цветов самих кирпичей и промежутков между ними. Кроме того, нам еще понадобятся размеры брусков с учетом имеющихся промежутков и процентное соотношение общих размеров с реальными размерами кирпича (**рис. 1**). Итого получается четыре переменные:

```
uniform vec3 BColor1, Bcolor2; // (1.0,0.4,0.0), (1.0,1.0,1.0)
uniform vec2 Bsize, BP; // (0.30,0.15), (0.90,0.85)
```

Чтобы определить, к какой точке объекта принадлежит текущий фрагмент, нам необходимо передать в varying-переменной координаты рассчитываемой вершины. Для этого в vertex-шейдере служит строка

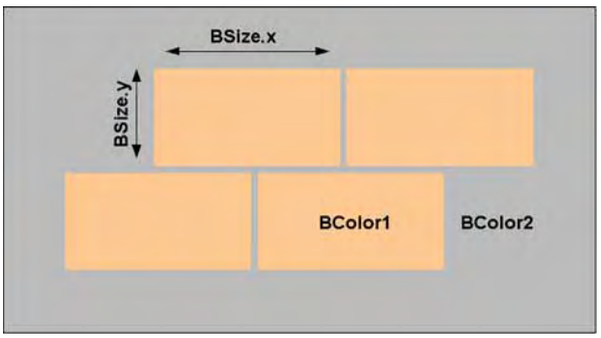


Рис. 1. Выглядит не очень, но это же просто чертеж!

```
MCposition = gl_Vertex.xy;
```

Таким образом мы также удалим нежелательный эффект скольжения рисунка. Не забудьте разместить эту переменную в обоих шейдерах!

Теперь нам нужно определить, как рисовать текущий ряд кирпичиков: ведь каждый второй у нас идет со смещением. Однако вначале необходимо вычислить номера ряда и блока в нем. Простым делением координат вершины на размер кирпича мы получим ряд в переменной Lpos.y и номер блока в Lpos.x:

```
vec2 Lpos = MCposition / Bsize;
```

А вот теперь самое интересное! Для определения смещения мы воспользуемся оператором ветвления IF...ELSE. Замечу, что правила написания его абсолютно аналогичны таковым в C/C++. Так как у нас кирпич должен смещаться на 0.5 от своей длины, то для расчета мы возьмем номер ряда из переменной Lpos.y, умножим его на 0.5 и отбросим функцией fract целую часть результата. Полученный результат сравнивается с 0.5. Таким образом, через раз условие будет истинным, а ряд – смещаться на 0.5:

```
if (fract (Lpos.y * 0.5)>0.5) Lpos.x +=0.5;
```

Для выбора цвета на текущем участке необходима функция, возвращающая 1 на кирпиче и 0 на промежутке. Их у нас будет две, ведь существуют и горизонтальные, и вертикальные компоненты. Результаты обеих функций перемножаются для получения окончательного цвета.

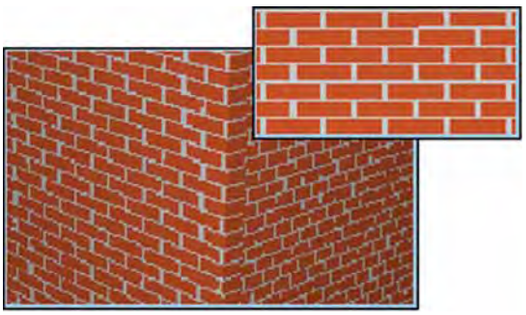
И снова на помощь нам приходит мощь языка GLSL в виде уже готовой функции step (edge, x). Вообще-то она служит для получения импульсов прямоугольной формы. В нашем случае, функция будет возвращать 0 при условии BP.x <= Lpos.x и единицу в случае несоответствия. Однако вначале следует вычислить координаты фрагмента по отношению к активному блоку:

```
Lpos = fract (Lpos);
vec2 uses = step (Lpos, BP);
```

Дело осталось за малым – надо вычислить окончательное значение цвета и умножить его на переменную outLight для получения правильного освещения фрагмента. Воспользуемся встроенной функцией mix (vec4 x, vec4 y, float a). Ее воплощение выражается формулой x*(1.0-a)+y*a. Так как результат операции uses.x * uses.y может принять либо единицу, либо ноль, то и функция будет возвращать одно-единственное значение (при 1 – цвет кирпича, при 0 – цвет промежутка).

Конечный код фрагментного шейдера кирпичной стены (**рис. 2**) таков:

```
//GLSL fragment shader
uniform vec3 BColor1, BColor2;
uniform vec2 BSize, BP;
varying vec2 MCposition;
varying float outLight;
void main() {
    vec2 Lpos = MCposition / BSize;
    if (fract (Lpos.y * 0.5)>0.5)
        Lpos.x +=0.5;
    Lpos = fract (Lpos);
    vec2 uses = step (Lpos, BP);
    vec3 color = mix (BColor2, BColor1, uses.x * uses.y) * outLight;
        gl_FragColor = vec4 (color, 1.0);
```



› Рис. 2. Почти как настоящий кирпич.

```
}
```

Noise-фактор

Зачастую при упоминании термина «шум» у непосвященных возникает ассоциация либо со звуком, либо с отсутствием сигнала на телеэкране (хаотичный «снег»). Однако в трехмерной графике использование «чистого» шума не имеет смысла. Ведь, как правило, текстура при наложении на объект размножается, и в случае «чистого» шума добиться необходимого узора невозможно. Шумовые функции выдают непрерывную псевдослучайную последовательность, которая зависит от начального значения. Благодаря этим особенностям появляется возможность имитации цветов природных и искусственных материалов, неровностей поверхностей, «хаотичной» анимации.

В случае с шейдерами предлагается три способа работы с шумом:

» Использование встроенных функций GLSL;

» Текстурные карты;

» Функции, определяемые пользователями.

Язык программирования шейдеров имеет встроенную процедуру для генерации шумов – noise(). Использование этого подхода имеет ряд преимуществ по сравнению с остальными. Во-первых, не расходуется текстурная память на хранение карты шумов. Кроме того, шейдеры, использующие noise(), не зависят от конкретного приложения. Однако в наше время эта функция, как правило, не применяется из-за неполной аппаратной реализации. Гораздо проще и быстрее один раз вычислить необходимую последовательность и сохранить ее в текстурной карте. В каждой текстуре можно хранить до 4 различных значений и получать их за одно обращение. Данные разделяются и хранятся в каналах RGBA. Замечу, что могут использоваться не только двумерные, но и трех-, четырехмерные карты шумов. В нашем случае, мы будем работать с двумерной, уже заранее просчитанной картой. Что же касается процедуры генерации, то ее легко написать самому или воспользоваться всезнающим Интернетом. Здесь при создании карты шумов использовалась стандартная функция Перлина.

Попробуем для начала смоделировать узор, напоминающий мраморный камень. Исправьте вершинный шейдер, заменив в нем строку:

```
MCposition = gl_Vertex.xy;
```

на

```
MCposition = vec3 (gl_Vertex);
```

То есть, в отличие от первого шейдера, мы будем использовать полноценные трехмерные координаты вершины. Не забудьте при этом изменить описание переменной в начале программы. На этом модификация вершинного шейдера завершена.

Для фрагментного шейдера нам понадобятся следующие переменные:

```
uniform sampler2D NoiseMap;
uniform vec3 MColor1; //0.80,0.86,0.74
uniform vec3 MColor2; //0.35,0.15,0.1
```

где MColor1 хранит основной цвет камня, а MColor2 – цвет прожилок. Кроме того, шейдеру необходимо указать на текстурную карту, которая будет использоваться для генерации шума. За это отвечает строка uniform sampler2D NoiseMap.

Решение задачи донельзя простое. Загружаем в вектор значения шума с шагом 1.5 для активного пикселя:

```
vec3 noisevec = texture2D (NoiseMap, MCposition *1.5);
```

Вычисляем интенсивность узора путем сложения обоих векторов:

```
float intensity = (noisevec[0] - 0.5) +(noisevec[1] - 0.25);
```

Для создания прожилок воспользуемся функцией sin (float x):

```
float ssin = sin(MCposition.y * 100.0 + intensity * 100.0)+0.4;
```

Осталось вычислить конечный цвет способом, который мы уже использовали для шейдера кирпичной стены. Но слишком уж это скучно! Давайте усложним задачу, разбив мрамор на некое подобие плиток. Чтобы не изобретать велосипед, воспользуемся готовым шейдером кирпича с небольшими модификациями.

Конечный результат работы программы должен отвечать следующим требованиям:

» Поверхность модели разбита на квадраты, напоминающие шахматную доску;

» Узор мрамора непрерывен и неразрывен для всей площади объекта.

Исходя из поставленных условий, логичнее всего произвести разбивку на квадраты, а потом залить все узором мрамора. За рисунок кирпичей у нас отвечали строки

```
if (fract (Lpos.y * 0.5)>0.5)
        Lpos.x +=0.5;
Lpos = fract (Lpos);
vec2 uses = step (Lpos, BP);
```

Опытным путем несложно добиться, чтобы небольшая модификация кода привела к желаемому результату:

```
if (fract (Lpos.y )>0.5)
        Lpos.x +=0.5;
Lpos = fract (Lpos);
vec2 uses = step (Lpos, BP);
```

Однако результат выбора цвета необходимо занести в отдельную переменную и обобщить только после вычисления кода «мрамора». Конечный код фрагментного шейдера мраморных плиток выглядит следующим образом:

```
//GLSL fragment shader
uniform sampler2D NoiseMap;
uniform vec3 MColor1;
uniform vec3 MColor2;
uniform vec3 BColor1, Bcolor2; //0.0,0.0,0.0 | 0.5,0.5,0.5
uniform vec2 BSize, BP; //1.30,1.15 | 0.50,1.0
varying float outLight;
varying vec3 MCposition;

void main ()
{
        // вычисляем квадраты
vec2 Lpos = MCposition / BSize;
if (fract (Lpos.y )>0.5)
        Lpos.x +=0.5;
Lpos = fract (Lpos);
vec2 uses = step (Lpos, BP);
vec3 color1 = mix (BColor2, BColor1, uses.x * uses.y);

// генерируем узор
vec3 noisevec = texture2D (NoiseMap, MCposition *1.5);
float intensity = (noisevec[0] - 0.5) +(noisevec[1] - 0.25);
float ssin = sin(MCposition.y * 100.0 + intensity * 100.0)+0.4;

// перемешиваем и загружаем в gl_FragColor
vec3 color = mix(MColor2+color1, MColor1+color1, ssin) * outLight;
gl_FragColor = vec4 (color, 1.0);
}
```

У полученного шейдера немало недостатков, однако со своей ознакомительной функцией он вполне справляется (рис. 3). Если еще немного поиграть с параметрами, можно добиться гораздо более естественного результата, но нас ждет последний в этом уроке шейдер деревянной поверхности.

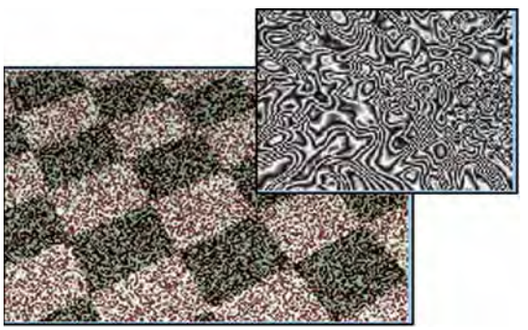


Рис. 3. Результат работы шейдера мрамора (контрастность усилена).

Под фанеру

Принцип его создания примерно такой же, как и в случае с мрамором. Уточним условия задачи:

- Рисунок поверхности состоит из чередования светлых и темных участков в форме концентрических цилиндров;
- Для реалистичности необходимо искажение цилиндров при помощи шума.

Для шейдера деревянной поверхности нам понадобятся переменные, содержащие основной и дополнительный цвета, масштабирование и фактор искажения для создания шума:

```
uniform vec3 LWood; //0.9,0.8,0.6
uniform vec3 DWood; //0.6,0.3,0.04
uniform vec3 NScale; //0.1,0.1,0.1
uniform float NNess; //3.0
```

Как обычно, вначале загружаем в вектор карту шумов и настраиваем необходимый тип генерации:

```
vec3 noisevec = vec3 (texture2D (NoiseMap, MCposition * NScale)*
Nness);
```

Последние два параметра отвечают за толщину и узор волокон. Поиграйте с ними и посмотрите, что получится. Мы высчитываем конечные координаты в соответствии с полученным вектором и координатами вершины:

```
vec3 location = MCposition + noisevec;
```

Теперь займемся рисованием кругов. Для этого нужно знать расстояние от оси бревна до текущего годичного кольца. Возведем координаты x и z в квадрат и извлечем из них квадратный корень. Количество кругов можно контролировать, если перемножить коэффициент масштабирования на полученный результат:

```
float dist = sqrt (location.x * location.x * location.z * location.z);
dist *=4;
```

Однако то, что у нас сейчас получилось, очень мало напоминает дерево. Обычно структура древесины не состоит из строгих концентрических цилиндров с резким разделением на темные и светлые полосы, а имеет плавные переходы между ними. Добиться этого можно, если написать функцию с постепенным изменением значения от 0 до 1.0 и наоборот. Необходимо сложить полученное расстояние с тремя октавами шума и выделить из результата дробную часть. Небольшой кусок кода будет отвечать за своевременное изменение конечного значения:

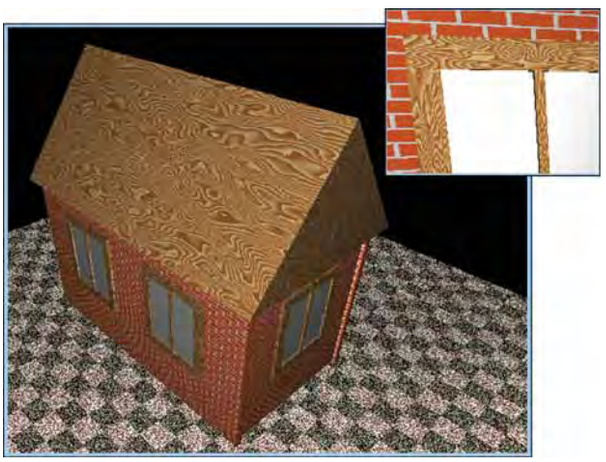


Рис. 4. Дом, который построил Дж... Нет уж, мы!

```
float r = fract (dist + noisevec[0] + noisevec [1] + noisevec[2]);
if (r > 1.0) r = 2.0 -r;
```

Вот таким образом мы получили необходимые данные для функции mix на выходе. Остальной код вам хорошо знаком. В целом, фрагментный шейдер дерева выглядит следующим образом (рис. 4):

```
//GLSL fragment shader (wood)
uniform sampler2D NoiseMap;
uniform vec3 LWood;
uniform vec3 DWood;
uniform vec3 NScale;
uniform float NNess;
varying float outLight;
varying vec3 MCposition;

void main ()
{
    vec3 noisevec = vec3 (texture2D (NoiseMap, MCposition * NScale)*
NNess);
     vec3 location = MCposition + noisevec;
    float dist = sqrt (location.x * location.x * location.z * location.z);
     dist *=4;
    float r = fract (dist + noisevec[0] + noisevec [1] + noisevec[2]);
    if (r > 1.0)
                r = 2.0 -r;
    vec3 color = mix (LWood, DWood, r);
    color *= outLight;
    gl_FragColor = vec4 (color, 1.0);
}
```

Процедурные текстуры занимают немаловажное место в реалистичной визуализации сцены. Они компактны, шустры и очень гибки в плане настраиваемости под конкретную задачу. Но даже если откинуть нарочитость простоты реализации наших шейдеров, видно, что до уровня естественности они явно не дотягивают. Большое значение при этом имеет правильная работа с освещением. Именно этому будет посвящен следующий урок. LXF

Время разбрасывать камни

История, согласно классикам, развивается по спирали. Наблюдая за тем, как происходит подключение школ к Интернету, **Галина Пожарина** делает вывод, что начался ее очередной виток.

Наш эксперт

Галина Пожарина
Директор Центра образовательных инициатив «Открытое ПО» (г. Санкт-Петербург), PhD/DCS

Как известно, организация доступа к Интернету большинства российских школ реализуется в рамках национального проекта «Образование» с 2006 года. Перечень подключаемых образовательных учреждений определялся в соответствии со ст. 12 Закона РФ «Об образовании» и типовыми положениями, утвержденными Правительством Российской Федерации. Он был согласован главами субъектов Российской Федерации, органами управления образованием, утвержден совместным приказом Мининформсвязи России и Минобрнауки России № 176/85 от 30 июня 2006 года. Данный перечень был уточнен совместным приказом Минобрнауки России и Мининформсвязи России от 31 января 2007 года № 38/14 «О внесении изменений в перечень образовательных учреждений, подключаемых к сети Интернет в рамках мероприятий ПНПО, утвержденный приказом от 30.06.06 № 176/85».

В течение двух лет с момента подключения (2007 и 2008 годы) оплата круглосуточного неограниченного доступа образовательных учреждений к сети Интернет на скорости не менее 128 Кбит/с осуществлялась за счет средств федерального бюджета. В дальнейшем вопрос об оплате доступа к сети Интернет должен решаться на уровне субъектов, которые обязаны обеспечить образовательные учреждения, планируемые к подключению к сети Интернет, автоматизированными рабочими местами (как минимум по одному на школу), а также осуществлять информационное сопровождение проекта.

Всего в рамках данного проекта к Интернету было подключено около 53 тыс. школ, что позволяет сделать вывод:

» Все российские школы теперь имеют доступ в Интернет.

По мере приближения к концу 2008 года, школы были поставлены перед следующим фактами:

» Компания «РТКомм.Ру», подключавшая их к Интернету в рамках национального проекта «Образование», предложила образовательным учреждениям перезаключить договор на оказание услуг интернет-доступа до 15 октября. В противном случае «РТКомм.Ру» прекратит оказание услуг, и школы могут оказаться отключены от доступа в Интернет.

» С 1 октября прекращается федеральное финансирование школ, подключенных к Интернету в рамках Национального проекта «Образование». Теперь доступ образовательных учреждений в Сеть будет оплачиваться из региональных бюджетов.

Таким образом, с 1 ноября оплата интернет-доступа за счет средств федерального бюджета будет прекращена для примерно семи тысяч школ. С 1 декабря финансирования лишатся еще две с половиной тысячи учебных заведений.

В соответствии с условиями нацпроекта, через два года после подключения школы к Сети регионы должны взять на себя обязательства по оплате интернет-доступа. Одновременно с этим региональные власти получат право выбрать компанию, которая предоставит школам выход в Сеть.

Однако в некоторых регионах при переходе на новую схему обслуживания доступа в Интернет в школах возникли проблемы. С одной стороны, не все регионы предусмотрели бюджетные средства на финансирование школьного Интернета. С другой, не во всех регионах был выбран новый подрядчик проекта – провайдер, который будет оказывать услуги доступа в интернет вместо «РТКомм.ру»

Перенос финансовой ответственности за организацию доступа школ к Интернету с федеральных источников на региональные, а с учетом финансовой самостоятельности образовательных учреждений, и на сами школы – процесс закономерный, понятный и естественный. Хотя вполне очевидно, что для ряда руководителей образовательных структур не только неприятный, но и весьма неожиданный. Надежды, что финансирование из средств федерального бюджета будет продолжаться, конечно, были. А теперь придется рассчитывать только на собственные силы и финансовые возможности.

Найдите пять отличий

В октябре 2007 года согласно Распоряжению Председателя Правительства РФ № 1447-р., Министерству образования и науки и Министерству связи выделено финансирование на решение следующих задач:

- Закупить лицензии на проприетарное программное обеспечение для всех школ сроком на три года;
- Обеспечить разработку и внедрение в школы свободного программного обеспечения по перечню программ, входящих в стандартный (базовый) пакет программного обеспечения.

В рамках проекта проведена пилотная апробация в 12 школах Республики Татарстан, 12 школах Пермского края и 10 школах Томской области.

Федеральное агентство по образованию выпускает циркулярное письмо «Об использовании ПСПО для общеобразовательных учреждений Российской Федерации» от 17 июня 2008 №15-51-450/01-09, в котором предлагает образовательным учреждениям из любого региона РФ принять участи в апробации ПСПО.

По планам Правительства РФ в 2009 году предполагается внедрение разработанного ПСПО во всех ОУ России с целью создания условий для альтернативного выбора программного обеспечения.

Интерес к проекту со стороны региональных и местных органов управления образованием достаточно невысок. Причина понятна: в школы поступила так называемая «Первая помощь» – пакет лицензионного программного обеспечения – в рамках реализации проекта «Обеспечение лицензионной поддержки стандартного базового пакета программного обеспечения для использования в общеобразовательных учреждениях Российской Федерации» внутри приоритетного национального проекта «Образование»

Проектом декларируется, что его реализация позволит уже в ближайшем учебном году и на перспективу в три года (до 2010 года) обеспечить все образовательные учреждения РФ лицензионным ПО, которое даст гарантированную возможность ведения образовательного процесса и административно-хозяйственной деятельности ОУ (http://shkola.edu.ru). Кроме того, проект направлен на поддержку института интеллектуальной собственности, усиления борьбы с «компьютерным пиратством», что особенно важно с учетом принятия 4-й части Гражданского кодекса РФ.

Закупка трехлетних лицензий на проприетарное ПО осуществлена для того, чтобы образовательные учреждения работали в этот «переходный период» на лицензионном (легальном) программном обеспечении, в то время как система образования будет подготовлена к переходу на СПО.

Перевода школ на лицензионное программное обеспечение еще не произошло. Руководители образовательных учреждений и органы управления образованием еще не осознали до конца свою ответственность, а если и осознали, то действенных мер и алгоритмов перехода пока не выработано.

При этом полтора года из декларированных трех лет уже прошло. И к концу этого учебного года необходимость принять ответственное решение ляжет на плечи директоров школ, методистов, информатиков. Не окажется ли школа перед опасностью не отключения от Интернета, а отключения от информационных технологий вообще?

Дискутировать, на мой взгляд, уже поздно. Необходимо учиться, апробировать и внедрять. Внедрять открытое и свободное программное обеспечение. LXF

» Кто
Владимир Георгиевич Слугин, начальник вычислительного центра Нижегородского радиотехнического колледжа (бывший ГРЭТТ – Горьковский радиоэлектротехнический техникум).

» Откуда
г. Нижний Новгород

» Проект
Модернизация программного обеспечения компьютерного парка колледжа.

» Историю успеха записал Евгений Балдин

От программистов — программистам

Технические специалисты – та категория людей, которая должна хорошо представлять себе внутреннюю работу компьютера. Linux здесь – естественный выбор, что подтверждают слова **Владимира Слугина**.

Наш колледж занимается обучением студентов по специальностям, связанным с программированием, базами данных, web, программированием микроконтроллеров, проектированием электроники и локальных вычислительных сетей, а также по бухгалтерским специальностям, связанным с работой в программном пакете «1С:Бухгалтерия».

Компьютерный парк в нашем учебном заведении составляет без малого 200 машин, и 80% из них используется в учебном процессе (задействованы в лабораториях и в вычислительном центре), а остальные 20% применяются в дополнительных подразделениях организации. Компьютеры играют важнейшую роль на этапе практического обучения студентов. Кроме того, в вычис-

лительной технике нуждаются такие службы, как, например, бухгалтерия.

После принятия в начале 2008 года поправок к ГКРФ стало совершенно ясно, что в стране назревает новая груда проблем, с которой прежде всего столкнутся рядовые пользователи и, в особенности, системные администраторы, на плечи которых ляжет ответственность за использование программных продуктов. Дело Поносова «подлило масла в огонь» разгорающихся дискуссий о законности применения программ непосредственно в нашей организации.

Основных причин, по которым мы выбрали именно свободное программное обеспечение, насчитывается несколько:

» Мы не платим деньги за использование свободного программного обеспечения и имеем право копировать дистрибутивы операционной системы и программ на носители студентов, преподавателей и даже просто посторонних людей.

» На наших серверах и компьютерах теперь отсутствуют вирусы, что избавляет нас от необходимости использовать антивирусное ПО, которое тратит машинные ресурсы и требует материальных затрат на приобретение лицензий, что при условиях эксплуатации большого парка машин играет немаловажную роль.

» Unix-подобные ОС, одной из которых является Linux, во всем мире считаются оптимальным выбором для использовании в качестве ресурсоемких масштабируемых и гибких систем, поэтому в организациях, связанных с ИТ, всегда востребованы специалисты в данной области. Это дает хорошую базу для последующего саморазвития дипломированного специалиста, вышедшего из стен нашего учебного заведения, и в будущем поможет ему получить хорошую и интересную работу по специальности.

» Гибкость, доступность и открытость исходных текстов системы интересна для сотрудников вычислительного центра, т.к. это позволяет нам реализовывать самые амбициозные планы по внедрению различных сервисов для удобства взаимодействия студентов, преподавателей и штатных сотрудников.

Примеры реализованных у нас сервисов, которые на сегодня стали возможными в результате перехода:

» Wiki – Локальная Википедия (свободная электронная энциклопедия), работающая на открытом движке MediaWiki.

» Контроллер домена LDAP – централизованная аутентификация студентов на любом из ПК любого вычислительного класса.

» IRC-сервис – организация общения подразделений колледжа, находящихся в локальной сети.

» Сайт колледжа и система тестирования с web-интерфейсом – это разработка наших выпускников.

» Различного рода распределенные системы с web-интерфейсом, автоматизирующие рабочие процессы сотрудников – тоже разработки наших выпускников.

Этот список пополняется с каждым годом, и уже есть планы на распределенные системы, сервис которых планируется предоставлять посредством локальной вычислительной сети.

Что же касается наших связей с компанией «Элсис» (см. стр. 30), то она пришла к нам сама. Результаты перехода и сотрудничества с ними мы оцениваем высоко, но хотелось бы остановиться именно на неразрешенных проблемах. Примерно 10% машин из имеющегося парка компьютеров перевести не удалось (это бухгалтерия и несколько рабочих мест, чья специфика работы завязана на взаимодействие с другими организациями города, которые «подсажены» на продукты Microsoft). Эти 10% работают полностью на лицензионном программном обеспечении, обычно Windows и 1С. Также на сегодня существует проблема использования в Linux таких приложений как «Компас 3D», «1С:Бухгалтерия» (для обучения бухгалтерским специальностям). Сейчас эта проблема решается параллельным использованием ОС Windows по лицензии MSDN для учебных заведений, позволяющей устанавливать копию Windows на неограниченное число ПК, но *только для использования с целью обучения студентов*. То есть в результате у нас настроена гетерогенная сеть, где каждый ПК имеет две операционные системы. А вот все серверы работают только под управлением Ubuntu Server 8.04 [данная система использовалась в колледже еще до начала масштабного перехода, – *С. Бессонов*].

Также хотелось бы отметить, что на мой взгляд (здесь я, наверное, выражу мнение всех преподавателей нашего вычислительного центра), очень большая проблема состоит в отсутствии русскоязычной документации по широкому спектру программных продуктов. Это очень сильно мешает их внедрению в школы и такие организации, как наша.

В заключение, хотелось бы пожелать всем оправданного использования программных продуктов, неважно, каких: свободных или коммерческих. Главное – стремиться к их правомерному применению в каждой отдельно взятой ситуации и вырабатывать культуру обращения с различного рода программными средствами. Сначала в отдельно взятой семье, затем в отдельно взятой школе, институте, государстве, в конце концов. Так как таковая культура у нас в стране только начинает становиться, то, по моему мнению, самое время принять участие в ее развитии, чем мы и занимаемся и вам желаем. LXF

» Историю успеха записал Евгений Балдин

» **Кто**
Антон Евгеньевич Панченко, начальник отдела ИТ администрации Черниговского района Приморского края.

» **Откуда**
Черниговский район, Приморский край.

» **Проект**
Перевод администрации на открытое программное обеспечение.

СПО районного масштаба

Администрация – это, безусловно, не школа и не вуз, но различий между ними меньше, чем кажется на первый взгляд. Об этом свидетельствует и личный опыт **Антона Панченко**.

По образованию я инженер-строитель-технолог, закончил Хабаровский государственный технический университет, но по специальности не работал, так как почти сразу пошел по стезе ИТ-специалиста. Общий стаж – около 16 лет, работал в ИТ-отделах военного комиссариата, пенсионного фонда, федерального казначейства. Теперь уже почти четыре года работаю в администрации Черниговского района Приморского края.

Первое мое знакомство с GNU/Linux началось с двух дистрибутивов: FREESCO 0.2.7 и Mandrake 8.2. Первым был восхищен, а второй не впечатлил. В принципе, понятно, почему это произошло, ведь назначение у этих систем разное, а я все оцениваю с точки зрения ИТ-специалиста.

FREESCO расшифровывается как FREE ciSCO – это бесплатная и свободная замена коммерческим маршрутизаторам, поддерживающая до 10 Ethernet/ARCnet/Token Ring/ARLAN-сетевых карт и до 10 модемов и никакого отношения к пресловутой SCO Group, как можно было подумать по названию, не имеющая. FREESCO приятно удивил своим соотношением объема дистрибутива и функционалом.

Mandrake на тот момент мне не понравился. Были проблемы с русификацией, X-сервером и тому подобное. Возможно, я просто не сумел его грамотно настроить. После этого я пробовал еще несколько RPM-дистрибутивов (ASP, Red Hat). Чуть позже проникся красотой FreeBSD и до недавнего времени только ее и использовал. Сейчас же на моем рабочем месте стоит Mandriva PowerPack Spring

x86_64, а дома Ubuntu Hardy Heron AMD64. Оба дистрибутива меня полностью устраивают.

Причина, по которой производился переход на свободное ПО, в общем, одна – это необходимость исполнять федеральные и иные законы. С января текущего года, как известно, вступили в действие поправки к федеральному закону «Об авторском праве и смежных правах», а также к Гражданскому и Уголовному кодексам, ужесточающие наказание за использование нелицензионного ПО. Естественно, риск попасть в неприятное положение в этом случае увеличивается.

В мою зону ответственности попадает примерно 100 компьютеров, включая серверы. Все Intel, AMD, VIA x86 – никакой особенной экзотики, хотя все модели от Celeron 400/512 до Core2Duo и Xeon присутствуют.

На данный момент в качестве базового дистрибутива у нас используется Mandriva PowerPack 2008.1. У Mandriva имеется центр разработки и поддержки в России, что удобно. Кроме того, Linuxcenter.Ru предлагает приобретать лицензии, которые ограждают пользователей Mandriva от претензий со стороны проверяющих органов, не всегда понимающих особенности лицензирования операционных систем. Бумага с печатями на них действует умиротворяюще.

Хотелось бы остановиться на вопросе технической поддержки. Та услуга, которая входила в стоимость дистрибутива, оказалась не очень полезной: я обращался к ней всего пару раз. Но если рассматривать миграцию как разновидность поддержки, то ее достоинства сложно переоценить. Услуги по самой миграции с Windows на Mandriva выполняла местная компания ООО «Информационные ресурсы» из г. Владивостока [филиал «ГНУ/Линуксцентр – Дальний Восток», – *прим. ред.*]. Специалисты выезжали к нам и несколько недель упорно трудились. В конечном итоге я получил готовый результат, где все «просто работало».

Фактически нужно было параллельно выстроить новую инфраструктуру с учетом имеющихся информационных систем, серверного и настольного оборудования и требований к информационной безопасности, а потом постепенно вывести из строя старую. Для этого был составлен и, в процессе переговоров, утвержден рабочий план.

В результате мы подняли PDC (Primary Domain Controller) на *Samba*, установили сервер *OpenLDAP* и «прикрутили» к нему *DNS*, прокси-сервер *Squid* с web-интерфейсом и редиректором *SAMS* (SQUID Account Management System – программное средство для администрирования доступа пользователей к прокси-серверу *Squid*), почтовый сервер *Exim* – все с авторизацией в LDAP. Сейчас пробую установить и настроить Mandriva Directory Server (централизованный сервер LDAP для управления учетными записями, почты, доступа к файловому серверу и прокси) – этакий продукт «все в одном». Для Windows-приложений (*Консультант+*, *Cuneiform* и другие) мы приобрели и установили WINE@Etersoft. В тех случаях, когда он оказывается бессильным, используем родные приложения Windows в режиме сервера терминалов.

Следует отметить, что мы и до перехода всегда старались использовать свободное или, в крайнем случае, бесплатное программное обеспечение. Все серверы, кроме сервера терминалов на Windows, работают под управлением Linux либо FreeBSD. Там, где не нужен полноценный *MS SQL*, применяется *MS SQL Express Edition* – на сегодня одним Linux, к сожалению, не обойтись. Например, есть несколько рабочих мест, связанных с использованием электронных цифровых подписей, а там Linux не работает. Так что некоторое количество лицензий на рабочие места и сервер терминалов Windows пришлось купить.

До процесса миграции было издано распоряжение по администрации об использовании СПО. В такой ситуации быть недовольным, конечно, никто не запрещает, однако все равно приходится работать с новой операционной системой. Обучение ведется практическими методами. Теория, к сожалению, результатов не дает. Стараемся разъяснять специфические для дистрибутива нюансы. Недовольство проявляется в основном оттого, что пользователям трудно отвыкнуть от интерфейса и поведения Windows. Возникают проблемы и по причине непривычной пользователям иерархии файловой системы, и из-за невысокой скорости работы KDE. Понятно, что можно использовать легковесный оконный менеджер (*Xfce*, *Fluxbox* и т.п.), но мы хотели унифицировать рабочие окружения. Какое-то время не монтировались USB-брелки (вылечилось обновлением ОС), не заработал как надо один сканер и пара принтеров. И, конечно же, есть *OpenOffice.org*. *OOo* – это вообще тема для отдельного разговора. Работает вроде все, но и не очень приятных мелочей хватает. Все эти проблемы решаются, конечно, обновлением и руками, но требуют внимания и терпения.

Основной плюс от такого смелого шага – это, несомненно, экономия средств местного бюджета. Стоимость ПО от Microsoft, даже по самым скромным подсчетам для нашей администрации, должна было составить примерно 1 миллион 200 тыс. рублей. И это не считая продуктов Adobe, Abbyy, Symantec и других. Стоимость Linux-решений в нашем случае, с учетом всех работ, оказалась в шесть раз меньше даже такой заниженной оценки. В некоторых ситуациях миграцию можно вообще выполнить, не отдав ни копейки «на сторону». Разумеется, в этом случае потребуются гораздо большие усилия со стороны структурного ИТ-подразделения. Для себя я бы отметил еще удобство администрирования: SSH, локальный репозиторий, несколько скриптов, и – «вуаля»: можно легко обновлять рабочие станции и выполнять рутинные операции. Не стоит также обходить вниманием повышенную (по сравнению с Windows) безопасность Linux, обусловленную архитектурными особенностями. Ну и сами свободы СПО приятно греют душу.

По результатам перехода в июне при поддержке Департамента связи и информатизации Приморского края в администрации Черниговского района проходил краевой семинар по вопросам использования СПО в органах местного самоуправления. Мне на почту пришло несколько писем из разных регионов страны. Так что, возможно, подобных «историй успеха» прибавится.

Мне нравятся идеи СПО. Каких-то конкретных планов пока нет, но я хотел бы в ближайшем будущем заниматься информационно-коммуникационными технологиями с использованием свободного ПО. К сожалению, как всегда, не хватает главного – времени, но я с этим пытаюсь бороться.

В заключение хочу сказать, что главная сила СПО – это сообщество. Основную помощь получаешь, общаясь в IRC, форумах, изучая списки рассылки и читая всевозможные man и HOWTO. От себя хотелось бы пожелать сообществу использовать СПО везде, где это возможно: ставить GNU/Linux знакомым; участвовать в проекте по его внедрению в школы; переводить документацию; писать в блоги о СПО. Все пойдет в плюс, а большие дела всегда состоят из множества совместных малых. LXF

› Сотрудница администрации за рабочим компьютером.

Ответы

Есть вопрос по Open Source? Пишите нам по адресу: answers@linuxformat.ru!

В этом месяце мы отвечаем на вопросы про:

1. DVD-риппинг и Mencoder
2. IMAP/POP
3. Двойную загрузку
4. Почтовые агенты
5. Сканирование
6. Беспроводные сети
7. Разбивку диска
8. Потоковые медиа
9. Монтирование NFS

1 Кино для Eee

Я в восторге от моего Eee, и я с большим интересом читаю ваши учебники. Хотелось бы смотреть на Eee DVD-фильмы во время поездок. Можно, конечно, купить внешний DVD-привод, но тогда пропадет ультрапортативность, поэтому хотелось бы записывать фильмы на SD-карты. Обязательно ли использовать 4-гигабайтную карту для каждого фильма, или фильмы можно перекодировать, уменьшив размер?

Пол Джеффрис [Paul Jeffries]

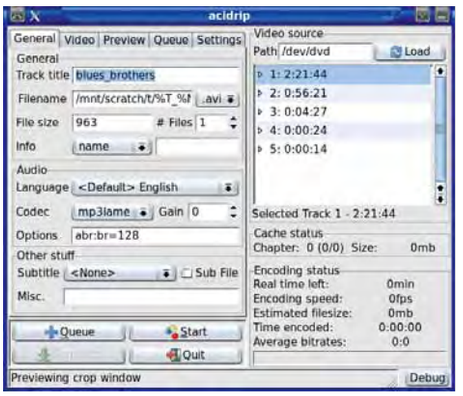
Обычно это делается так: фильм с DVD копируется в видеофайл, который можно записать на SD и смотреть на Eee с помощью *SMPlayer*. Единственный минус – вы теряете DVD-удобства (субтитры, меню, дополнительные звуковые дорожки и пр.). Препарировать DVD можно и на самом Eee, подключив внешний DVD-привод, но перекодирование видео такого формата круто грузит процессор и память, и может занять очень много времени. Лучше выполнить эту работу на более мощном ПК или ноутбуке.

Есть много программ для преобразования DVD в видеофайл; одна из лучших – *DVD Rip*, и она почти наверняка найдется в репозиториях вашего дистрибутива. Программа обвешана массой параметров, но большинство из них можно оставить по умолчанию. Главное – выбрать формат и битрейт для будущего видеофайла. С форматом все ясно: берите Xvid, он дает высокое качество при малом размере файлов и низкой ресурсоемкости. Битрейт определяет размер файла и качество воспроизводимой картинки. Чем меньше битрейт, тем меньше размер файла и хуже качество. *DVD Rip* может предложить вам указать битрейт и сообщит размер получаемого файла, либо сам назначит битрейт, исходя из объема свободного пространства, которым вы располагаете.

Хотя графический интерфейс *DVD Rip* весьма удобен, при обработке нескольких DVD суетиться с мышью уже не столь приятно. На помощь придет командная строка. *Mencoder*, входящий в комплект *MPlayer*, кодирует все, что *MPlayer* способен воспроизводить, включая DVD. Параметров у *Mencoder* так много, что его man-страница едва ли не рекордсмен по длине. Для установки параметров проще будет воспользоваться графическим интерфейсом *AcidRip* (http://untrepid.com/acidrip) и вызывать *Mencoder* автоматически. Закончив настройку, нажмите кнопку Export в панели Queue [Очередь], и начнется генерация скрипта **acidrip.sh** для вызова *Mencoder* с готовыми параметрами. Этот сценарий можно использовать повторно для каждого нового DVD, меняя только номер дорожки и имя выходного файла. Номер дорожки дается после **dvd://** при каждом вызове *Mencoder*. А можно отредактировать скрипт, чтобы номер дорожки и имя выходного файла указывались в качестве аргументов. Замените каждое вхождение номера дорожки (одно на проход, кстати, *Mencoder* работает лучше с двумя проходами) на $1, а имя выходного файла – на $2. Теперь скрипт можно запускать командой

```
sh acidrip.sh 1 mydvd.avi
```

$1 заменится на первый аргумент после имени скрипта, а $2 – на второй, и первая дорожка **mydvd.avi** будет скопирована в текущий каталог. Теперь можно копировать видео на SD-карту и смотреть где угодно. **MC**



AcidRip **можно использовать для риппинга DVD или для установки параметров** ***Mencoder*****.**

2 Пересылка почты

Статья Энди Ченнела в августовском номере (LXF108) порадовала меня: часто приходится работать с почтой в отъезде. У меня есть смартфон с Windows Mobile, на который я получаю почту. Работает хорошо, но маленький экран и убогая клавиатура меня уже достали. Я заказал Eee PC 901 с Linux и надеюсь, что вскоре работа с мобильной связью станет для меня более комфортной.

Но как настроить мобильную почту на моем будущем Eee? У меня есть общеизвестный почтовый адрес bob@bob.net, но я не хочу перетаскивать все контакты

Наши эксперты

Мы найдем эксперта по любому вопросу – от проблем с установкой системы или модемом до сетевого администрирования; главное – спросить!

Нейл Ботвик
Владелец ISP и экс-редактор дисков для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.

Майк Сондерс
Майк был одним из создателей прототипа LXF – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.

Пол Хадсон
Пол – местный супер-программист, и он может и хочет управиться со всеми вашими проблемами по части web и баз данных.

Грэм Моррисон
Когда он не обозревает кучи программного обеспечения и не халтурит с MythTV, Грэм готов дать ответ касательно любого оборудования и проблем виртуализации.

Андрей Маркелов
Эксперт по решениям компании Red Hat.

Юлия Дронова
Когда ее компьютер не занят выполнением команды emerge, Юлия использует его, чтобы модерировать ЛинуксФорум.

КУДА ПОСЫЛАТЬ ВОПРОСЫ:

Пишите нам по адресу: answers@linuxformat.ru или спрашивайте на форуме: www.linuxforum.ru

на какой-нибудь **bob3876@googlemail**, я хочу сохранить свой **bob.net**. С какой стати мне его забросить и пользоваться стандартным Gmail/Yahoo/AOL-ящиком? Сейчас мой смартфон оставляет копии сообщений на сервере, и я могу снова прочесть их, вернувшись домой. Мне надо сохранить эту возможность, но еще лучше было бы иметь две синхронизированных копии почты: на Eee и на домашнем ПК. Очень бы не хотелось, чтобы одна часть сообщений оказалась только на настольном компьютере, а другая – только на Eee. Можно ли добиться этого с помощью IMAP? Еще хотелось бы, когда Eee нет с собой, получать доступ к своей почте с другого компьютера через web-почтовик. Возможна ли синхронизация всех почтовых клиентов – домашнего, мобильного и web? И нельзя ли сделать так, чтобы мои письма, откуда бы они ни были отправлены, приходили к адресатам от **bob@bob.net**? Или я слишком много хочу?

Питер Брогден [Peter Brogden]

Конечно же, IMAP все это умеет. В отличие от POP, задуманного просто для перекидывания сообщений с почтового ящика провайдера на ваш компьютер, IMAP – протокол непосредственной работы с удаленным почтовым сервером (хотя он может и сохранять локальные копии на случай, если вы окажетесь вне сети). С IMAP вопроса о сохранении сообщений на сервере просто нет: они именно там и находятся, и доступны для чтения откуда угодно. Единственный способ удаления сообщений – команда с клиентского компьютера. Вопрос о синхронизации тоже отпадает: все компьютеры «держат» IMAP-сервер за общее хранилище данных и обращаются с ним одинаково.

Как упоминал Энди в своей статье, единственный недостаток IMAP в сочетании с ящиком провайдера – ограничение объема хранилища, плюс морока со сменой электронных адресов при переходе к другому провайдеру.

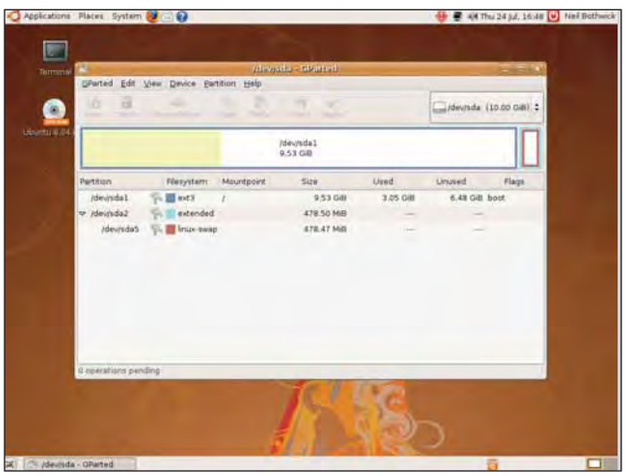
Последней неприятности можно избежать, обзаведясь собственным доменным именем (сейчас это не так уж дорого). Один из способов снятия ограничений объема – настройка собственного IMAP-сервера, при наличии постоянного подключения к Интернету и круглосуточно работающего компьютера. Все это мы подробно объясняли в LXF99, но – вкратце: вам нужно установить *Fetchmail* и *Procmail*, чтобы получать почту от провайдера через POP, и настроить их созданием файла **.fetchmailrc**

```
set daemon 300
poll mail.myisp.com with proto POP3
      user ‘myispuser’ there with password
‘mypass’ is ‘myuser’ here options keep
mda ‘/usr/bin/procmail -d %T’
```

Настройте *Procmail* на доставку почты, поместив в **~/.procmailrc** вот это:

```
MAILDIR=/var/spool/mail
DEFAULT=$MAILDIR/$LOGNAME/
```

и поставьте файл в автозагрузку с помощью **~/.kde/Autostart**, менеджера сеансов Gnome или того, что предусмотрено у вас – почта будет загружаться с ящика провайдера и храниться локально (за это отвечает параметр keep). Теперь нужен IMAP-сервер вроде Dovecot. Установите его и отредактируйте **/etc/dovecot.conf** заменой строк



Запустите редактор разделов Ubuntu в Live-режиме, и пускай кроит.

```
#listen= [::]
#mail_location =
```

на

```
listen = *
mail_location = /var/spool/mail/%u
```

С помощью менеджера служб установите запуск Dovecot при загрузке и проверьте работу, настроив почтовый клиент на подключение к **localhost**. Когда все заработает, у вас может возникнуть необходимость удаленного подключения к своей машине. Чтобы обойти динамическое распределение адресов провайдером, зарегистрируйте доменное имя на **dyndns.org**. Осталось настроить маршрутизатор на работу с портами 143 и 993 (143 – стандартный IMAP, 993 – безопасный IMAP). **НБ**

3 Начнем с начала

Я сам собрал компьютер и установил Ubuntu. Но что бы я ни делал, войти в систему не могу. Не могли бы вы подсказать мне, как отформатировать винчестер Maxtor SATA? Я хотел бы начать с нуля и организовать два раздела – один под Vista, другой под Linux. Пробовал изучать команды командной строки, да они какие-то непонятные. Я хочу полностью отформатировать диск и установить обе системы заново. Пробовал команды оболочки, но безуспешно.

Гленн Стеррит [Glenn Sterritt]

О Vista, и Ubuntu обладают средствами для полного форматирования жесткого диска перед установкой. Vista или другую Windows следует устанавливать первой, так как Ubuntu опознает Windows-систему и включает ее в меню загрузки. Vista же бесцеремонно перезапишет любую «соседку». Можно установить Windows и отдать ей весь диск. Затем, запустив установщик Ubuntu, предоставить ему урезать владения Windows и выделить место под Ubuntu. Но вмешательства в файловую систему Windows можно избежать. Лучше загрузиться с установочного диска Ubuntu и выбрать вариант Try Ubuntu Without Making Any Change To Your Computer [Попробовать Ubuntu без внесения изменений в компьютер]. Затем запустите System > Administration > Partition Editor [Система > Администрирование > Редактор разделов] и удалите все разделы, затем создайте один раздел в начале диска, под Windows. Установите для него любой размер, а остаток диска »

Краткая справка про...

Ссылки

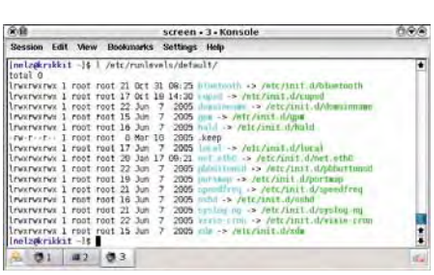
Не те, что в браузере, но тоже сберегают время и место.

Ссылка – это способ представить один и тот же файл одновременно в двух местах, или под разными именами. Можно просто скопировать файл в несколько разных мест, но у такого способа есть пара недостатков. Во-первых, загромождается хранилище, а во-вторых, если исходный файл обновляется, на копии это никак не влияет. Файловая система Linux обладает средствами для доступа к одному файлу или каталогу из разных мест: ссылки. Ссылки бывают двух типов – жесткие и символьные (symlink).

Технически, любой файл – это жесткая ссылка: ведь его имя привязано к структуре данных на жестком диске, где хранится содержимое файла. Говоря о жестких ссылках, мы часто имеем в виду дополнительные ссылки, когда файл имеет несколько имен. Основной недостаток жестких ссылок в том, что они могут быть связаны только с файлами, а не с каталогами, и все они должны находиться в пределах одной файловой системы. Символьные ссылки обладают гораздо большей гибкостью, и в любой Linux-системе их полно. Просмотреть ссылки можно с помощью параметра **-l** команды **ls**, да и файловые менеджеры такие ссылки показывают. Классический пример мы видим в **/usr/src**, **/usr/src/linux** там – это символьная ссылка на исходный код текущего ядра (скажем, linux-2.6.15). Таким образом можно установить больше одного ядра, а любая программа, которой необходимо обратиться к исходным текстам, заглянет в **usr/src/linux** – и не ошибется. Множество симлинков вы найдете в **/etc/rc.d**. Создать симлинк можно командой

```
ln -s /path/to/file /path/to/link
```

Один или оба пути могут быть относительными.



В /etc символьных ссылок всегда много, ведь эти файлы должны быть доступны отовсюду.

оставьте пустым. Не пытайтесь создать Linux-раздел. Нажмите кнопку Apply [Применить], и пусть этим занимается редактор разделов.

Теперь перезагрузитесь, замените диски и установите Windows в только что размеченный раздел (Windows видит его как **C:**). Как только Windows установится и заработает, перезагрузитесь с диска Ubuntu и снова выберите вариант работы без изменений для компьютера, но на этот раз запустите установку, щелкнув значок Install [Установить] на рабочем столе. Пускай установка продолжается в свободный раздел, который вы создали прежде (это стандартное поведение инсталлятора), только прикажите создать отдельный **/home**. По окончании работы установщика перезагрузитесь, и вы увидите меню загрузки с выбором между Windows и Linux.

Все это можно сделать из командной строки, причем гораздо быстрее. Но все графические инструменты на установочном диске есть, и вам незачем изучать всякие «непонятные» команды. Не так уж они и непонятны, просто нужно к ним попривыкнуть. А графические инструменты именно для того и придумали, чтобы нам с вами было удобнее… **ГМ**

4 Почтовые агенты

ВЯ сейчас разбираюсь с устройством почтового сервера Linux. Я знаю, что агент передачи почты (mail transfer agent, MTA) – это программа вроде *Sendmail* или *Postfix*, которая реально осуществляет отправку, получение и т.п. Знаю, как работают обычные почтовые клиенты типа *Evolution*. Но я хочу уяснить, как именно работает агент доставки, например, *Procmail*. Обязательно ли устанавливать его на моей машине?

Jellyman_4eva, из форумов

ОКогда кто-нибудь отправляет письмо на ваш адрес, MTA дает разрешение на подключение удаленному серверу и получает сообщение. Аналогично работает обратная операция: когда вы отправляете через него почту, агент находит первый сервер в цепочке и передает ему сообщение. Получив письмо, MTA, как правило, доставляет его в ваш почтовый ящик. Многие агенты делают это самостоятельно, особенно в простом случае добавления почты в почтовый ящик локального пользователя. Но чаще этим занимаются отдельные агенты доставки (delivery agent, MDA). Как видите, без MDA можно обойтись, но лучше этого не делать, по нескольким причинам. MDA типа *Procmail* может быть очень полезен – ведь он умеет обрабатывать сообщения, прежде чем предъявлять их вам: например, отбрасывать громоздкие приписки сообщений списков рассылки или раскладывать письма по отдельным почтовым ящикам. *Procmail* способен также управиться с сообщениями об отсутствии (делайте это после фильтрования списков рассылки, иначе, вернувшись из отпуска, обнаружите себя выкинутым из какой-нибудь рассылки за нарушение регламента). И все это делается еще до доставки почты пользователю.

MTA, MDA, MUA: почта продирается сквозь дебри акронимов.

Еще одна причина, по которой стоит обратить внимание на программы вроде *Procmail* – наряду с глобальными правилами в **/etc/procmailrc**, каждый пользователь может установить собственные правила в **.procmailrc** своего домашнего каталога. На многопользовательской системе это дает каждому пользователю больше свободы. Если вы захотите перенаправлять сообщения на адрес Gmail ва время отсутствия, для этого достаточно составить несложное правило.

Возможен и такой вариант, когда MTA передает почту другим программам, не MDA (например, антивирусу или спам-фильтру), и только потом в игру вступает MDA. В таком случае сообщение поступает от MTA на фильтр спама, на антивирусную проверку, к MDA, и только потом попадает в почтовый ящик пользователя. Для сбора и чтения сообщений через POP3 или IMAP нужна еще одна программа, которая извлекает письма из почтового ящика, доставленные туда MDA. **НБ**

5 В поисках сканера

ВЯ установил Ubuntu 8.04 на Dell Vostro 1500. Если подключаю свой старый сканер Snapscan E50, *XSane* видит его, но выдает такое сообщение:

```
Error: “Failed to open device ‘snapscan
libusb:003:006’: Invalid argument.”
```

Однако, если я подключаю сканер к ПК с Windows, а затем возвращаю обратно в Dell, *XSane* включается и

Часто задаваемые вопросы

Руткиты

С вашей машиной все в порядке?

Итак, что такое руткит?
Это комплект ПО, который вредоносная программа устанавливает с целью скрыть свое присутствие в системе: например, модификации *ls* или *ps*, которые не показывают процессов вредителя. Обычно руткиты скрыты, как троянцы, чтобы не возбуждать подозрений. Название образовалось потому, что это ПО может устанавливать (и управлять им) только пользователь root. Пораженный компьютер иногда называют ‘rooted’ (да не обидятся жители Австралии).

А насколько они опасны?
Задача руткита – прикрыть деятельность главного злодея. Обычно это делается за счет изменения программ, наблюдающих за состоянием компьютера – так, что они не замечают ничего особенного. В этом отношении руткиты отличаются от троянцев.

А троянцы – это что?
Троянские кони – сокращенно троянцы – вредоносные программы, маскирующиеся под нечто безвредное. Название происходит от способа, которым воспользовались древние греки, чтобы проникнуть в город Трою. Отсюда фраза: «Бойся данайцев, дары приносящих… и файлообменников, предлагающих ломаное ПО». Adware, которое, к счастью, для Linux-пользователей не проблема, один из примеров таких программ: они «звонят домой», не спросив разрешения хозяина.

Имеет ли это отношение к музыкальным CD?
Лишь недавно проприетарное ПО от Sony было признано руткитом, хотя оно и для Windows. Программа была установлена на огромное количество компьютеров, причем пользователей даже не поставили об этом в известность и не спросили их согласия. Это самый знаменитый руткит!

А как узнать, что мой компьютер заражен?
Есть разные программы, способные выявить в системе наличие руткитов. Работать они могут двумя методами. Одни из них отслеживают характерные последовательности кода известных руткитов, другие сравнивают контрольные суммы MD5 с теми, которые были записаны на «чистой» системе. Во втором случае, очень важно установить и запустить такую программу сразу же после установки ОС (чтобы подсчитать MD5 на нетронутой системе). Программы такого типа – *Rootkit Hunter* с **www.rootkit.nl**, и *Chkrootkit* с **www.chkrootkit.org**. Запускать их нужно регулярно, лучше с *Cron*. Это позволит программам «пробуждаться» при входе в Интернет, чтобы обновлять базы данных известных руткитов.

И они вылечат мой компьютер?
Нет. Если уж руткит установлен, компьютер раз и навсегда теряет доверие. Под прикрытием, вредоносная программа делает что хочет, и надежно заметает свои следы; попорченным может быть что угодно. Единственный выход – сделать резервные копии данных и файлов конфигурации и переустановить систему.

» работает безупречно. Стоит выключить питание сканера – проблема возникает опять. Вот мне и приходится всякий раз переставлять USB-кабель из настольного компьютера в ноутбук, чтобы сканер заработал. Чего не хватает моей Ubuntu-машине?

Петер Калебка [Peter Kalebka]

О Если устройство работает только после подключения к Windows, можно быть уверенным: для него необходима подгрузка прошивки [firmware]. Во многие устройства, особенно коммуникационные, прошит не весь необходимый программный код. Некоторая – возможно, большая – часть кода содержится в firmware-файле, который подгружается драйвером при первом включении устройства, прошивка же содержит лишь то, что необходимо для получения и использования подгруженного кода, который остается в устройстве до тех пор, пока не будет выключено питание. Поэтому, когда вы переносите сканер от компьютера к ноутбуку, не выключая питание, он продолжает работать. Зачем изготовители делают это? А так для них дешевле и удобнее. Можно иметь несколько вариантов прошивки для соблюдения законов и правил разных стран (вместо разработки специальных модификаций оборудования). Особенно широко это применяется на устройствах для беспроводных сетей, ведь в каждой стране свои ограничения по мощности, частотным диапазонам и пр. Обновление тоже упрощается, ведь достаточно заменить только драйвер. Для пользователей Windows проблем нет: все контролируется драйвером, при обновлении драйвера автоматически устанавливается и новое firmware.

Почти так же дело обстоит и на Linux – главное, знать про необходимость firmware-файла. Подробно об этом можно прочесть на **http://snapscan.sourceforge.net**, но если коротко, то вам нужен файл **snape50.bin**, установленный в Windows-разделе. Скопируйте его куда-нибудь в Linux-систему (**/lib/firmware** – по умолчанию) и отредактируйте **/etc/sane.d/snapscan.conf**, добавив строчку

```
firmware /lib/firmware/snape50.bin
```

Для этого понадобится отдать команду *sudo*, а путь в **snapscan.conf** должен соответствовать реальному местоположению файла. Теперь драйвер *Sane* будет знать о файле firmware и подгружать его, устранив еще одну зависимость от Windows. Если вы установите и обновите драйвер на Windows, можно скопировать новый firmware-файл. Но если старый нормально работает, можно и не копировать... **НБ**

6 Беспроводной ребус

В Как заставить мой маршрутизатор D-Link (DSL-2640B) и адаптер D-Link (DWL-111) работать под Linux (OpenSUSE 10.3)? Сомневаюсь, что сопроводительная документация к Windows может помочь. Дело в том, что, если я изменю настройки, все это больше не будет работать на Windows. А еще: что должно быть совместимо с Linux – маршрутизатор или адаптер?

Йоханнес Ланг [Johannes Lang]

О Маршрутизатор подключается к Интернету, поэтому его настройки трогать нельзя, на любой операционной системе или компьютере. А вот адаптер соединяет компьютер с маршрутизатором, и поэтому должен иметь подходящий драйвер для работы с Linux. К сожалению, номера модели недостаточно, чтобы порекомендовать подходящий драйвер для вашей карты: изготовители имеют привычку менять установленные внутри чипсеты, оставляя прежнюю нумерацию. Я сам поймался однажды, купив USB-адаптер D-Link вроде вашего: они поддерживались драйвером Linux prism, а оказалось, что недавно произошел переход на новый, неподдерживаемый чипсет.

На случай, если для вашего чипсета не найдется Linux-драйвера, есть обходной путь, но сначала попробуйте найти родное решение. Весьма вероятно, что в вашей карте стоит чипсет Ralink RT73. OpenSUSE 10.3 включает драйвер для таких чипсетов, но не устанавливает его по умолчанию. Перейдите в раздел Управление ПО [Software Management] *Yast* и установите **rt2x00-kmp-default**. Теперь включите сетевой адаптер и перейдите в раздел Информация об устройствах [Hardware Information] *Yast*, там же. Если ваше устройство появилось в разделе Сетевые карты [Network Card], переходите в раздел сетевых карт и приступайте к настройке. Если устройства нет, отключите его от компьютера, откройте терминал и наберите:

```
su
tail -f /var/log/messages
```

Подключите адаптер и понаблюдайте за выводом, проносящимся на терминале (Ctrl+C остановит вывод). Запустите

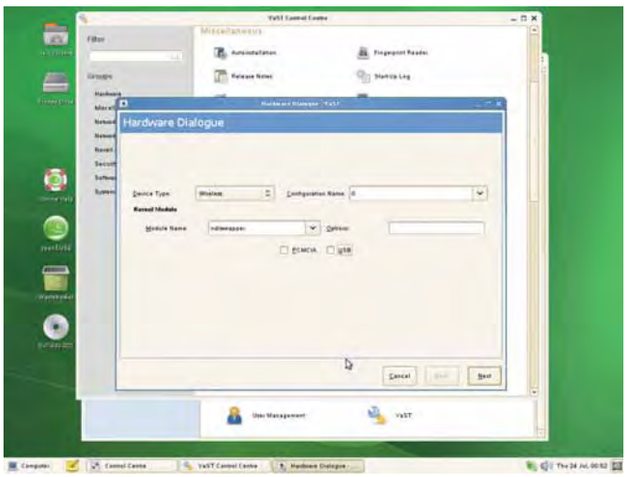
```
modprobe rt73usb
```

и если не получите ответа – значит, установлен «правильный» драйвер. Переходите в раздел сетевых карт *Yast* и настраивайте подключение. Возможно, понадобится указать модуль драйвера (rt73usb).

Если все оказалось напрасно, есть возможность использования Windows-драйверов, с помощью пакета под названием *NdisWrapper*. Это эмуляция интерфейса сетевых драйверов Windows. Установите пакет с помощью *Yast*. Поройтесь в CD из комплекта адаптера и найдите там файл с расширением **.inf**, затем в терминале (он ведь еще открыт) наберите

```
ndiswrapper -i /путь/к/driver.inf
modprobe ndiswrapper
```

Первая команда установит драйвер туда, где он обнаруживается *NdisWrapper*; вторая загрузит модуль *NdisWrapper* и, если все прошло гладко, обойдется без сообщений. Теперь создавайте интерфейс в *Yast* с помощью кнопки Добавить [Add], устанавливайте Тип устройства [Device Type] на Беспроводное [Wireless], в качестве Имя модуля [Module Name] укажите NdisWrapper, оставив все остальное без изменений, и продолжайте обычный процесс настройки. **НБ**



› ***Yast*** **настроит любую сетевую карту – только укажите ему, где находится драйвер для нее.**

7 Нарезка диска

В В Linux я новичок. Я разделил жесткий диск своего ноутбука на пять разделов: два по 50 ГБ формата ext3 (для **/** и **/home**), два NTFS по 50 ГБ и 30 ГБ, и 1 ГБ под **/boot**. Хотел бы переработать свой дистрибутив (у меня Ubuntu 8.04), чтобы убрать домашний раздел с hda3 и переместить его на hda1 (корневой) без переустановки системы. Я заметил, что в корневом разделе тоже есть каталог **/home**. Эти файлы – дубликаты пользовательских файлов раздела **/home** (hda3)?

Еще один вопрос: существует ли скрипт для установки беспроводной карты Realtek RTL8187B под Linux? Я пробовал гуглить и понял только, что мне нужно использовать Windows-драйверы во время инсталляции. Не можете ли порекомендовать Linux-драйвер для моей карты?

Тоедсит Нетратанавонг [Toedsit Netratanawong]

О Домашний каталог корневого раздела – это точка монтирования вашего отдельного раздела **/home**. В отличие от Windows, где каждый раздел представляет собой отдельную букву диска, Linux монтирует все устройства на основную файловую систему. Домашняя директория корневого раздела пуста, ее единственная функция – обеспечить доступ к файлам на домашних разделах. Поэтому дубликатов там нет, только оригиналы.

Перенести файлы с отдельного раздела на корневую файловую систему, конечно, можно, но это плохая идея. Мы получаем множество писем с просьбой рассказать, как провести обратную операцию. Если ваш домашний каталог находится на отдельном разделе, операционную систему можно в любой момент переустановить, а все ваши файлы и настройки останутся нетронутыми. Если вы все же решили продолжать, загрузите дистрибутив (например, *Knoppix*) с Live CD- или DVD. Ваши разделы будут смонтированы раздельно, например **/media/hda1** и **/media/hda3**. Теперь можно переносить файлы с **media/hda3** на **/media/hda1** (вы убедитесь, что там и в самом деле пусто), но сначала хорошенько подумайте – так делать не рекомендуется.

вы могли бы сделать весьма полезную перепланировку, так как при нынешней разбивке диска теряется очень много места. Правда, это задача нелегкая. Лучше всего воспользоваться для этого *QtParted* с «живого» *Knoppix* или *GParted* с подобного диска Ubuntu. Корневому разделу 10 ГБ много: большинство систем и половины этого не используют. Для **/boot** за глаза хватит 50 МБ, ведь все, что там есть – это пара ядер и загрузчик (у меня в разделе **/boot** меньше 12 МБ). Настоящие пожиратели пространства – это разделы **/home**, ведь там хранятся все персональные

Вопрос-победитель (английская версия)

Баланс параметров

Удаленный диск (rw) на моем сервере через NFS монтируется клиентскими компьютерами во время загрузки с помощью *fstab*. При каждой перезагрузке сервера, кажется, необходимо, чтобы клиентские компьютеры тоже перезагрузились или выполнили монтирование заново. На моем Amiga я мог бы сделать это, отдав клиенту команду **diskchange**. Можно ли делать то же самое на Linux, без перезагрузки клиентских компьютеров или перемонтирования вручную? Или хотя бы автоматизировать процесс?

Ричард де Ривас [Richard de Rivaz]

К счастью, перемонтировать устройство в Linux просто: наберите

```
mount /mount/point -o remount
```

Впрочем, кажется, в этом нет необходимости. NFS способна восстанавливать монтирование после «возвращения» сервера, и должна делать это по умолчанию. Проблемы возможны только при длительной перезагрузке (пять минут и более). Есть ли у сервера статический IP-адрес? Если используется DHCP, то после перезагрузки адрес может измениться. Но более вероятна ошибка при монтировании или экспорте файловых систем. Посмотрите журналы на серверной и клиентских машинах: обычно там содержатся ценные указания. Проверьте параметры экспорта на сервере командой

```
exportfs -v
```

в особенности установки **sync** или **async**. А может быть, дело в параметрах **soft** или **intr** на клиентских машинах.

Страница man для *exportfs* покажет все возможные параметры для сервера, а man для NFS – для параметров монтирования. **Intr** часто предпочитается **soft**, так как в этом случае снижается вероятность потери данных при перезагрузке сервера. Альтернативный параметр, используемый по умолчанию – **hard**. Все эти параметры определяют поведение клиента при отсутствии ответа сервера. Так как вариант soft может вызвать потерю данных на сервере, его лучше использовать при экспорте в режиме «только чтение». В стандартной ситуации лучше всего применять **intr**: в этом случае клиент продолжает отвечать на сигналы, а не блокируется намертво. Лучше всего испробовать различные варианты – может быть, какой-то из них окажется для Вас более приемлемым. Лично я предпочитаю бороться с проблемой таймаутов с помощью **intr**. **НБ**

Технология счастья

SUNRADIO.RU

сетевое радио под ключ на базе Linux
новое будущее вашей компании
pr@sunradio.ru +7 812 955 76 70 www.sunradio.ru

данные (вот почему не рекомендуется прицеплять их к операционной системе) и Windows-разделы, по этой же причине.

Драйверу *NdisWrapper* для работы необходим Windows-драйвер, но сейчас драйвер для работы с вашей картой входит и в состав ядра Linux. Чтобы избежать конфликта между драйверами, перейдите в менеджер пакетов *Synaptic* и посмотрите, не установлен ли в системе *NdisWrapper*. Если есть, удалите, и тогда ваша карта должна появиться в разделе сетевых устройств после перезагрузки. **ГМ**

Нужна помощь!

› Для наилучшего ответа на ваш вопрос нам нужно знать как можно больше подробностей. Детально опишите конфигурацию системы. Если вы получили сообщение об ошибке, приведите текст сообщения и точно опишите вызвавшие его действия. Если у вас проблемы с оборудованием, то опишите его. Если Linux уже запущен, то выполните в root-терминале следующие команды и прикрепите к письму файл **system.txt**:

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

› Пожалуйста, помните, что сотрудники журнала НЕ являются авторами или разработчиками Linux, любых пакетов или дистрибутивов. Зачастую люди, отвечающие за приложения, выкладывают большую часть информации на web-сайты. Попробуйте почитать документацию!

Мы стараемся ответить на все вопросы. Если вы не нашли ответ на свой, это, возможно, потому, что мы уже ответили на похожий вопрос.

Большой вопрос Как настроить трансляцию потокового медиа по сети?

8 Школьный сервер

Я администрирую школьную инфраструктуру на основе Linux (PCLinuxOS/Karoshi). Хотелось бы настроить медиа-сервер с web-интерфейсом, чтобы учителя могли использовать видеоматериалы из NAS. Не могли бы вы подсказать подходящее решение из области открытого ПО?

Eddygrinder, из форумов

Возможно, вам подойдет *MediaTomb* с **www.mediatomb.cc**. Он присутствует в нескольких дистрибутивах, а для PCLinuxOS есть неофициальные пакеты на **http://hack.mypclinuxos.com**. После установки нужно прежде всего решить, каким образом *MediaTomb* будет хранить данные. При первоначальном запуске *MediaTomb* создает файл **~/.mediatomb/config.xml**. Отредактируйте его, указав для **sqlite3_enabled** или **mysql_enabled** значение 'yes' или 'no'. В некоторых дистрибутивах используется глобальный файл конфигурации **/etc/mediatomb/config.xml**. Если вы выбрали *SQLite*, при первом запуске *MediaTomb* создаст файл базы данных, но для крупных коллекций лучше подойдет *MySQL*-сервер. Чтобы воспользоваться *MySQL*, нужно создать базу данных с именем *MediaTomb* и заполнить ее, набрав

```
mysql -p mediatomb </usr/share/mediatomb/mysql.sql
```

Затем создайте учетную запись пользователя с паролем и правом чтения/записи для базы данных *MediaTomb*. Если вы знакомы с клиентом командной строки *MySQL*, воспользуйтесь им, или прибегните к помощи *PHPMyAdmin*, так будет проще. Отредактируйте файл конфигурации, установив корректные имя пользователя и пароль.

Запустите *MediaTomb* и загрузите **http://localhost:49152** в *Firefox* (с Konqueror не работает). Если вы администрируете сеть с другого компьютера, используйте вместо этого адрес сервера. Вы увидите перед собой пустое окно; нажмите кнопку +, чтобы начать ввод содержимого. Добавляя видеофайл, определите для него тип Item, введите заголовок, под которым он будет отображаться, и Location – абсолютный путь к файлу. Немаловажна установка MIME-типа: один из способов определить правильное значение – использовать для видеофайла команду

```
file -i /path/to/video.avi
```

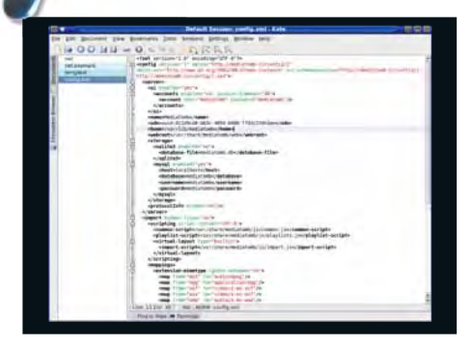
Каталоги на сервер можно добавлять в режиме отображения файловой системы, а редактировать их свойства – в режиме отображения базы данных, чтобы активировать автоскан. Это означает, что можно иметь один или более каталогов, автоматически выдающих *MediaTomb* помещенный в них контент.

Некоторые программы UPnP (universal plug and play) позволяют просматривать и воспроизводить контент сервера, а для Linux-компьютеров этот контент можно сделать доступным любой программе. Установите *DJmount* с **http://djmount.sf.net** (для этого необходима установленная файловая система Fuse). Создайте подходящую точку монтирования – скажем, **/mnt/media** – и смонтируйте

```
djmount /mnt/media
```

Теперь в каталоге **/mnt/media** будет находиться каталог *MediaTomb*, плюс по одному каталогу для каждого UPnP-сервера сети, с подкаталогами для всех видов контента – например, **/mnt/media/MediaTomb/Video**. НБ LXF

Шаг за шагом: настройка MediaTomb



1 Начальная конфигурация

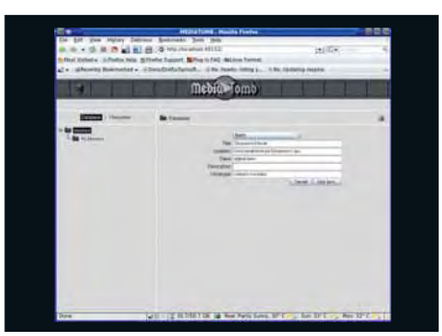
Отредактируйте **config.xml**, чтобы сделать выбор между *SQLite* и *MySQL*. Если выбираете *MySQL*, здесь же укажите имя пользователя и пароль.

2 MySQL попроще

Истинные хакеры при работе с базами данных отважно печатают длинные и загадочные командные строки, а когда их никто не видит, потихоньку открывают *PHPMyAdmin*.

3 С чистого листа

Поначалу *MediaTomb* имеет весьма бледный вид – эту пустоту вам предстоит заполнить контентом.

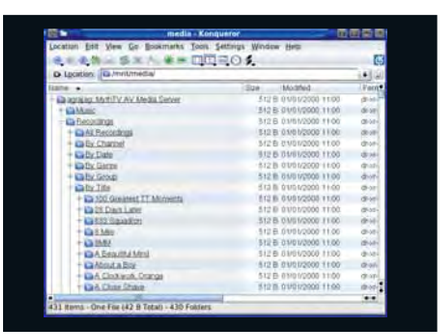


4 Добавьте первый файл

Чтобы добавить любой тип медиа-файлов, используйте параметр Item, потом запустите команду для файла и проставьте верный MIME-тип.

5 Добавьте контента

Каталоги можно поставить на автоматическое сканирование, и тогда любой медиа-контент, добавляемый в них, будет автоматически выкладываться на сервер.



6 Все есть файл

С помощью *DJmount* можно смонтировать любой UPnP-сервер, включая *MediaTomb* и *MythTV*, и обращаться к их контенту из любой программы.

Лучшие новинки открытого ПО на планете

LXF HotPicks

Ник Вейч
Серийный маньяк-взломщик ПО, бывший редактор **LXF**, Ник – хакер с незапамятных времен. Работает обычно в уютном командном пункте (сарае).

В ЭТОТ РАЗ ТОЛЬКО ДЛЯ ВАС: » Aria 2 » Brain Workshop » Diet Monger Ass Kicker » GNUjump » KleanSweep » Luke-SDK » NCFTP » Urssus »

Читалка новостей

Urssus 0.2.10

Версия 0.2.10 **Сайт** http://code.google.com/p/urssus

Новости – дело хорошее. Не будь новостей, все состарилось бы гораздо быстрее. Странно только, что люди хотят получать свои новости по-разному; потому-то, видимо, на свете столько новостных программ. *Urssus* – очередная из них. Но… минуточку: может, это хорошая программа? Не тот ли он самый движок, который вы искали всю свою жизнь – или ту часть своей жизни, которая прошла с момента появления RSS.

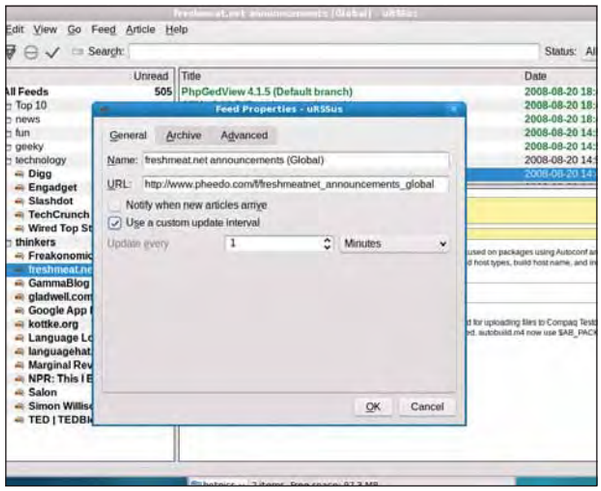
Написанный на Python, *Urssus* по-быстрому собирает для вас новости в сети и подает их в традиционном трехпанельном виде. Украшения сведены к минимуму, но это даже и кстати, если вам нужен ничем не перегруженный интерфейс. Хотя формально это приложение KDE, в нем нет ничего сугубо KDE'шного – в основном благодаря обращению напрямую к *Qt 4*. Применение кода рендеринга *Qt* было умным ходом, поскольку все выглядит именно так, как вам того хотелось бы.

Если ранее вы применяли другое приложение, *Urssus* сможет импортировать ваши новостные ленты. В нем также есть функция получения их списка с Google Reader, если вы до сих пор этим пользуетесь.

Запуск *Urssus* может оказаться непростым делом, поскольку он зависит от некоторых экзотических модулей Python. Программа установки проделывает отличную работу, отыскивая и устанавливая их, но на разных дистрибутивах она ведет себя по-разному. Попытка заставить ее работать на SUSE была сущим кошмаром, а вот с Fedora 9 проблем было очень мало. Не забудьте установить версию Python для разработчиков (как правило, это пакет *python-dev* в ближайшем к вам репозитории), да еще вам понадобится самая свежая версия *PyQt*: некоторые более старые пакеты были скомпилированы без модуля *WebKit*, на который опирается *Urssus*.

Предположим, вы его уже собрали (а если не удалось, то почему бы не обратиться на домашнюю страничку?); тогда вы, скорее всего, найдете его дружелюбным, чутким и простым в управлении.

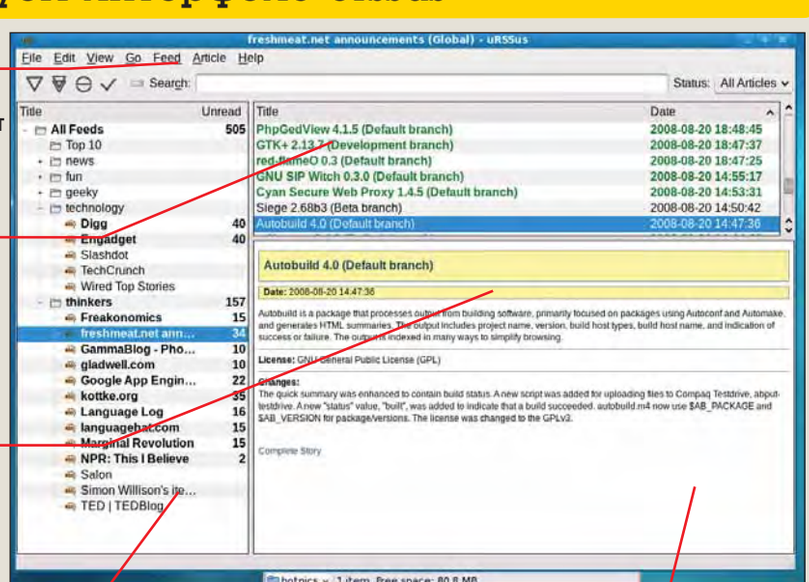
В *Urssus* есть кое-какие симпатичные функции, но порой настраивать его буквально мучительно – а если он рухнет, настройки придется удалить и начать все сначала. Определенно, на данный момент это скорее сарай для экспериментов где-то на задворках, чем кабинет для работы с важными материалами. Возможно, до *Akgregator* он пока не дотягивает, но поскольку это совершенно новый проект, с ним стоит познакомиться – а вдруг он вырастет в вашу любимую программу для чтения новостей?



› **Построенный прямо на *Qt*, *Urssus* не выглядит заядлым KDE'шником.**

> «Urssus по-быстрому собирает информацию в сети и подает ее вам.»

Исследуем интерфейс Urssus



Добавление новостных лент
Можете импортировать их и с Google Reader!

Заголовки
Отсортированы и окрашены в разные цвета для удобства навигации.

Просмотр сообщений
Эта панель содержит тело сообщения, отрисованное и стилизованное.

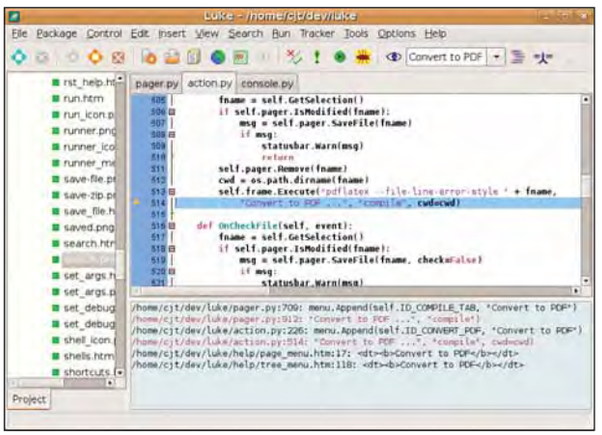
Управление новостными лентами
По этому дереву просто лазить.

Интерфейс сообщения
Если вам не нравятся стили, можете их изменить.

IDE

Luke-SDK

Версия 0.7.7.1 **Сайт** http://luke-sdk.berlios.de



Ищите – и обрящете; даже в живых системных файлах!

Наверное, не каждый день вы откидываетесь на спинку кресла, размышляя: «Да, знаете ли… этому миру действительно, действительно, ДЕЙСТВИТЕЛЬНО нужна еще одна многоязычная IDE». Возможно, такое бывает только по четвергам. Что ж, если вы читаете эти строки в четверг – ура! Ваши мечты сбылись! Вот вам еще одна многоязычная IDE.

Luke-SDK («используй исходник, Люк!», UTSL – ха-ха!) – это IDE на Python/*wxWidgets*, которая умудряется вместить уйму полезных функций при малых накладных расходах. Пускай это не *Eclipse* – но, с другой стороны, она и стремится не быть *Eclipse*: здесь нет никакой перегруженности, необходимости загружать раздутые среды Java, и т.д. и т.п.

Если вы хотите поработать с простым скриптом, web-страницей или файлом конфигурации, вы увидите, что *Luke-SDK* более чем подходит для этого. Поддержка проектов Java/PHP/Python/Ruby весьма солидная, и программа работает также с разнообразными вариациями C, JavaScript, Perl (а что, им кто-то еще пользуется?) и OCaml.

Налицо обычные функции – просмотр классов, быстрый просмотр ошибок, проверка синтаксиса, управление версиями, поиск… одна из самых лучших – возможность настроить IDE, чтобы она работала как вам нравится. Честно говоря, нередко именно это бывает главным. Немало разработчиков используют относительно «примитивные» инструменты типа *Vim* исключительно потому, что их легко заставить вести себя нужным образом. Пожалуй, *Luke-SDK* занимает место где-то посередине, предоставляя удобство гибкого GUI без перегруженности *Eclipse*.

Одна из самых удобных функций – живое обновление. *Luke-SDK* создана, чтобы поддерживать agile-разработку, что по большей части означает облегчение постоянного обновления и продолжительной работы с программами. У этой функции есть и несколько неплохих побочных эффектов, если вам надо открывать, редактировать или просматривать часто обновляемые системные файлы – например, файлы журналов…

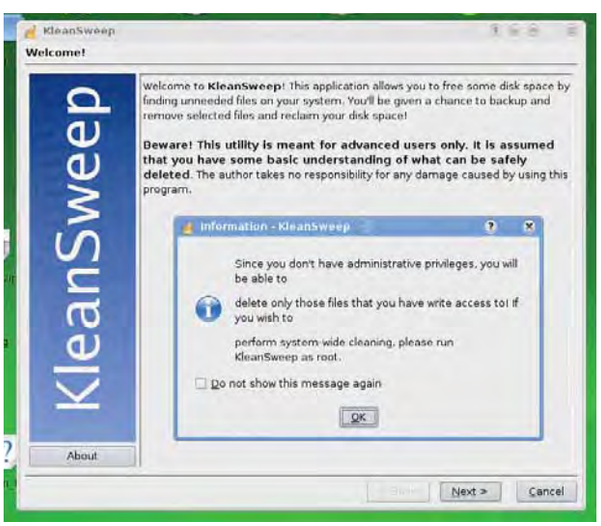
> «Luke-SDK сумела вместить много функций, не став перегруженной.»

Здесь нет никаких зависимостей, кроме свежей установки Python (2.5+) и самой последней версии *wxPython/wxWidgets*. Имеется даже удобный инсталлятор, с помощью которой установка промчится со свистом. Так чего ждать целую неделю до следующего четверга… знакомьтесь с ней прямо сейчас!

Программа удаления файлов

KleanSweep 0.2.9

Версия 0.2.9 **Сайт** http://linux.bydg.org/~yogin



Генеральную уборку запускайте из-под суперпользователя.

На вашем рабочем столе – беспорядок? Повсюду валяются пакетики от чипсов, старые наклейки с напоминаниями, ручки, которые перестали писать? Из слоя пыли выползают новые формы жизни, нагло тычутся в вашу клавиатуру и в итоге эволюционируют в редакторов диска Linux-журналов? Ну, тогда эта программа вряд ли вам поможет. Зато она берет на себя работу по очистке рабочего стола в вашем компьютере. В Windows есть встроенная в ОС утилита, подобная этой (*Disk Clean*), и, как ни больно нам это признавать, она в общем-то удобна. Перед вами ее вариация для KDE: она откапывает забытые файлы (ваши тоже называются **plop**?), осиротевшие символьные ссылки и объекты в кэше, а затем с вашего разрешения удаляет их.

Эта программа способна лихо смести целые группы файлов без всякой проверки, так что уж лучше сначала огласить весь список и снять выделение с тех ненужных файлов, которые вы бы все-таки сохранили. *KleanSweep* будет работать и от имени простого пользователя (тогда вашим плацдармом будет домашний каталог), и из-под root, хотя в этом случае стоит принять серьезные меры предосторожности. Учтите, иногда это единственный способ избавиться от мусора, забивающего всевозможные временные директории, вот и займитесь этим – но сначала скрестите пальцы на удачу, и будьте очень-очень внимательны и аккуратны.

Файлы-кандидаты на удаление появляются во вкладках, и между ними легко переключаться. Для нас самым полезным оказался раздел неиспользованных миниатюр. Выберите его, и вы найдете несметное количество эскизов скачанных изображений или ваших фотографий, которые занимают куда больше места, чем заслуживают. Безжалостно удаляйте их: они снова появятся при следующем просмотре папки! *KleanSweep* требует наличия библиотек KDE 3.3 и легко компилируется в большинстве систем; не исключено и наличие пакета именно для вашего любимого дистрибутива. Пользоваться им несложно, и *KleanSweep* изрядно поспособствует нормальной работе вашего рабочего стола – хотя бы в том, что легче станет пробраться к нужному файлу. Попробуйте…

> «…искореняет забытые файлы, ссылки-сиротки и объекты в кэше.»

Счетчик калорий

Diet Monger Ass Kicker

Версия 22 Август 2008 **Сайт** www3.telus.net/~jtstory/programs-dmak.html

Коли я есмь то, что я ем, то мир предстает в довольно жалком виде: это, например, значит, что сайт **LXF** приводится в действие Биг-Маком с картофелем фри. Цель *Diet Monger Ass Kicker* (или *DMAK*) – придать смысл бестолковому миру еды и подарить нам уверенность в том, что мы потребляем достаточно правильной пищи.

Наверное, одна из слабых сторон *DMAK* – некоторая сложность освоения, но ниже мы приводим краткий обзор, который должен вам в этом помочь. Итак, панель внизу справа отображает список продуктов (или ингредиентов). На панели слева содержится информация по энергетической ценности, наличию витаминов и прочей дребедени, потребной нашему телу для нормального функционирования (с точки зрения Департамента сельского хозяйства США). При выделении какого-либо продукта вся эта информация отображается в процентах (или в граммах, опционально) рядом с рекомендуемой суточной нормой потребления – так что вы можете подсчитать, содержит ли ваша пища все, что полагается.

Скорее всего, обнаружится, что вы потребляете больше, чем нужно – особенно жиров и сахара (нашему телу, похоже, вовсе не нужен сахар – да-да, для нас это тоже был сюрприз!), а отнюдь не недоедаете – если, конечно, ваша диета не состоит из дешевой еды в тех же количествах, что поглощал Морган Спарлок [Morgan Spurlock, американский режиссер, снявший фильм «Такой огромный я» о вреде фаст-фуда, – прим. пер.].

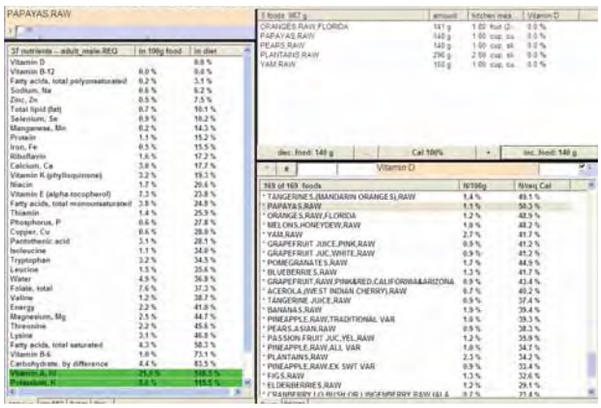
Конечно, интересно оценить свой рацион и увидеть, насколько все плохо, но все же главная цель данной программы – составление настоящей «диеты», по максимуму обеспечивающей все, что вам нужно, из того, что у вас есть. В программе имеются профили для людей разных типов (беременных женщин, младенцев и т.д.), и можно учесть особые требования к диете. Диеты и блюда рассматриваются *DMAK* как отдельные пункты, но это просто для удобства. Соедините несколько ингредиентов по рецепту определенного блюда, затем добавьте его в диету, чтобы взглянуть на показатели.

Синьор Помидор

DMAK использует принятые в США данные по энергетической ценности продуктов питания – а почему бы и нет? Вряд ли американский помидор сильно отличается от английского томата (надеемся, вы почувствовали разницу) по своей питательности [это если он не генномодифицированный, – прим. ред.]. Данные по энергетической ценности программа получает при установке, но при необходимости вы сможете скачать обновления вручную. Списки продуктов Сельхоздепартамента США отнюдь не исчерпывающи, зато широко предоставляют данные по продуктам, используемым в точках быстрого питания и закусочных. Хранятся данные в текстовых файлах ASCII, и можно добавлять к ним свои собственные, хотя над форматированием придется поломать голову. Есть некие тонкости в связях между файлами данных, и добавление всех лакомых продуктов требует побольше, чем просто усердия – может, лучше наплевать да вообще их не есть?

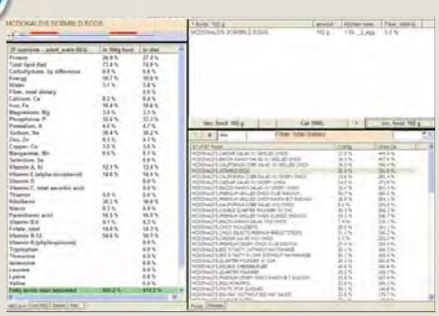
> «Diet Monger Ass Kicker придает смысл миру пищевых продуктов.»

Программа поставляется с собственным установщиком для Windows и Linux, и дает возможность попробовать также некоторые другие плоды творчества автора. Единственное настоящее требование здесь – это Python; все остальное добудет программа установки. И если вам удастся найти способ вести здоровый образ жизни, питаясь продуктом, сокрытым между двумя булочками с кунжутом, киньте нам об этом письмецо – на наш обычный адрес.



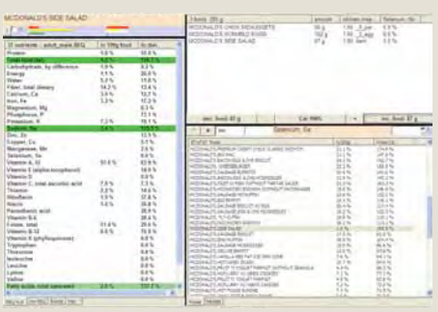
Бананы, кокосы… Апельсиновый рай… Посмотрите, что вы едите – если, конечно, хватит духу.

Шаг за шагом: Планируем свою диету



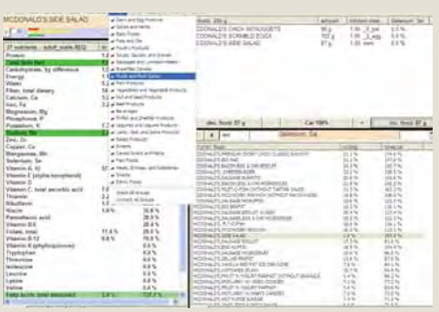
Добавим ингредиенты

Добавьте ингредиенты блюда – начните печатать и выбирать из списка, затем выберите кнопку «inc food», чтобы добавить порцию.



Создадим рецепт

По мере добавления новых ингредиентов, их энергетическая ценность появляется в левой панели.



Оценим свою диету

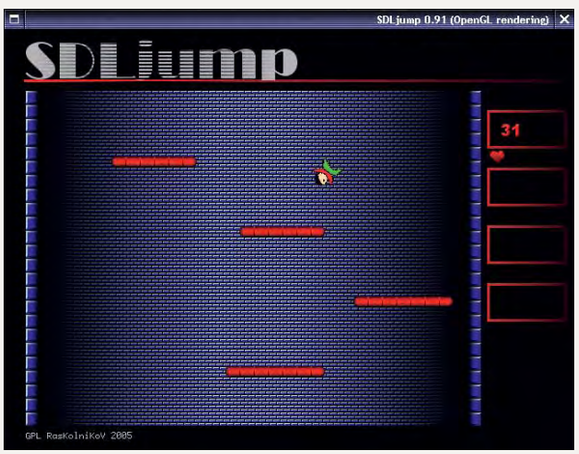
Сохраните полученные блюда, затем снова загрузите их, чтобы создать свою диету на каждый день.

HotGames Развлекательные приложения

Платформы

GNUjump 1.0.5

Version 1.0.5 **Web** www.gnu.org/software/gnujump



Прыгайте как лемминг, только в обратную сторону.

Представьте себе *«Гибель Посейдона»*. Теперь выкиньте из головы кадры, когда Памела Сью Мартин скидывает юбку, а Джин Хэкмен ныряет в море огня – и что у вас останется? Толпа людей, рвущихся срочно куда-то вскарабкаться. Вот и в *GNUjump* то же самое – это, если угодно, комбинация нарезки из телесериалов «Гладиаторы» и «Мир Марио». Прыгайте с платформы все время вверх, пока не рухнете на дно экрана. Задача эта из разряда Сизифовых трудов, поскольку единственной вашей наградой за адские усилия станет шанс понаблюдать, как платформы все быстрее уходят в небытие, требуя от вас прыжков как у лосося на речных порогах во время нереста, ради спасения своей жизни.

Кстати, а мы говорили, что у парнишки, которому приходится совершать все эти прыжки, похоже, подошвы с тефлоновым покрытием? Приземлиться на платформу не так уж сложно, совсем другое дело – удержаться на ней. Зато графика – яркая и блестящая, с хорошей анимацией. Не раз вам придется падать в пропасть, но утешением послужит отличающийся тонкой эстетикой и геометрической формой шлейф из пара, который потянется за вами. Впрочем, это утешение, наверное, будет единственным. Игра трудная и беспощадная. Наш основной совет – не останавливаться ни на мгновение, постоянно продолжая прыжки, а если вы промахнетесь, всегда есть шанс приземлиться на платформу внизу. Сделав паузу, чтобы попытаться выработать какую-либо стратегию, вы немедля превратитесь в котлету [в оригинале – Hovis, марка английского хлеба, – прим. пер.]. Так что не останавливайтесь. Сколько мы заработали очков? Хм-м, знаете, вы лучше сами с собой соревнуйтесь. Не хотим портить вам удовольствие…

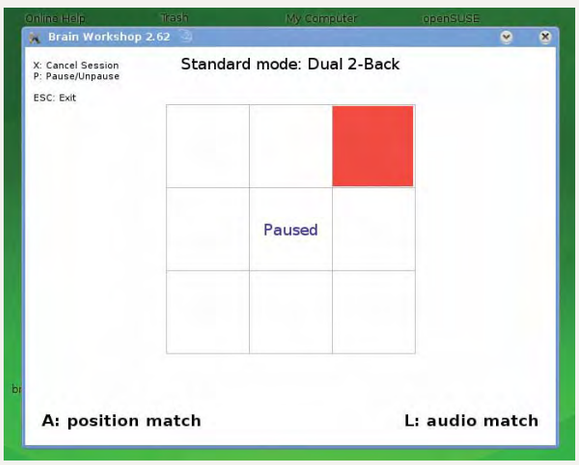
> «Сделав паузу, чтобы выработать стратегию, превратитесь в котлету.»

Игра использует библиотеки SDL – убедитесь, что они (а также соответствующие заголовочные файлы) у вас есть: их надо установить до попытки выполнить компиляцию. Вроде бы пакетов для *GNUjump* еще нет – вот вам повод сделать собственный; но код, кажется, в хорошей форме, и не должен доставить вам никаких неприятностей, так что откройте стр. 44 – и за компиляцию!

Головоломка

Brain Workshop

Версия 3.1 **Сайт** http://brainworkshop.sourceforge.net



Гордитесь своим умом? Эта игра заставит вас распроститься с заблуждениями…

Вы – из тех неженок, которые считают, что игра должна просто развлекать? Тогда вы явно обратились не по адресу. В цене сейчас мозги, и именно здесь вы и начнете платить по счетам.

Суть игры заключается в глазении на клетку с красной заливкой, расположенную в поле 3x3, и прослушивании названия буквы. Нужно всего лишь их запомнить. Боже, как просто! Затем вы видите еще одну красную клетку и слышите название еще одной буквы. Если клетка та же, вы нажимаете A, если звуки одинаковые, нажимаете L – вот и все. Реально просто! Но… это только на самом низком уровне – фактически, в обучающем режиме для умственно отсталых. В настоящем режиме придется запоминать буквы и расположения от двух ходов назад и более – да-да. Ну что, пригорюнились, умники?

Brain Workshop основана на очень нашумевшем исследовании – в документации и на сайте есть все цитаты, если вы до них охотник; а в итоге игра действительно развивает мозги. Мы испробовали ее на Майке, но особой разницы не заметили: он по-прежнему ест гамбургеры на обед день за днем.

Как вы уже догадались по очень простой концепции и минималистской игровой зоне, игра потребует более чем скромных ресурсов, были бы дисплей и звуковая карта. Никаких зависимостей, достойных упоминания, кроме Python. Не бойтесь инструкций, советующих открывать игру в IDE. Она прекрасно будет работать из оболочки, если вы прихватите python:

```
python brainworkshop.pyw
```

> «Brain Workshop основана на очень нашумевшем исследовании.»

Вначале игра кажется обманчиво простой, но довольно быстро превращается в очень, очень сложную. Сообщите, если за ночь ни с того ни с сего выйдете в эйнштейны.

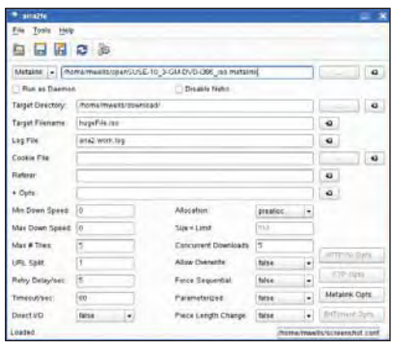
Утилита для скачивания

Aria 2

Версия 0.15.2 **Сайт** www.aria2.sourceforge.net

Большинство из нас даже не задумываются о менеджерах закачек, но *Aria 2* действительно делает нечто полезное. Работая в командной строке, вы можете просто указать ей URL для скачивания. Правильно используя некоторые из ее дополнительных опций, можно настроить загрузку в несколько потоков, позволяющую скачивать разные части файла одновременно (при условии, что сайт, с которого вы загружаете файл, это допускает). Все модули уже включены в нее, в том числе возобновление закачки. Она работает с прокси, она работает с FTP-прокси, она даже может использовать cookies (для авторизации) из *Firefox*.

Может даже показаться, что нет ничего, чего не смогла бы для вас сделать *Aria 2*. Тукс свидетель, она даже скачает файлы BitTorrent, если вы попросите. Это – инструмент командной строки, но имеется и несколько графических оболочек. *Aria 2* вообще-то не менеджер закачек. Она не так работает. На самом деле, это утилита, которая может сэкономить минуты – а то и часы – вашей жизни, потраченные на загрузку. Фишка в том, чтобы знать все



› ***Aria 2* – только для командной строки, но мы включили в LXFDVD также несколько сторонних GUI.**

волшебные команды. Самая полезная из них – переключатель **-s**, он управляет количеством потоков. Так,

```
aria2 -s3 http://www.someurl/somefile
```

запускает закачку в два соединения. Конечно, вы можете добавить и больше, но не забывайте, что большинство общественных FTP-сайтов ограничивает трафик до одного входа в единицу времени с одного и того же IP-адреса, и в этом случае использование множества потоков (или даже попытка это сделать) значительно замедлит темп.

FTP-клиент

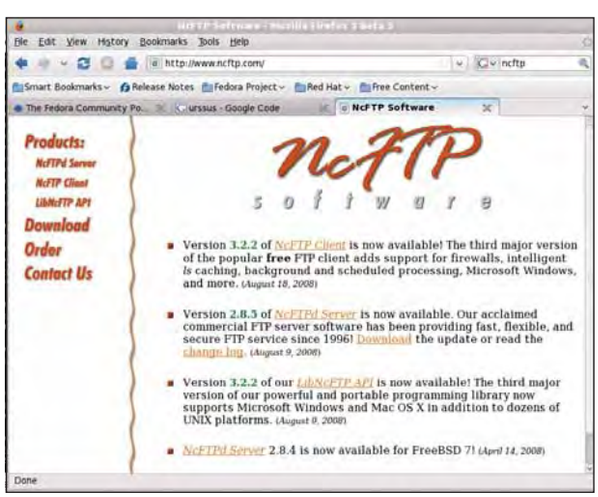
NCFTP 3.3.2

Версия 3.3.2 **Сайт** www.ncftp.com/ncftp

LINUX FORMAT HotPicks повторный визит

Не нужно быть суперкомпьютерщиком или даже пылать желанием им быть, чтобы использовать командную строку. Оболочка *Bash* – это здорово! Здесь можно сделать почти все, что угодно, не тратя времени на нажатие кнопок и дурацкие меню. Просто чудо, что идея графического рабочего стола вообще появилась на свет. Эти парни из Xerox, должно быть, действительно нечто. В любом случае, то, что вам надо использовать оболочку, вовсе не означает, что вы обречены работать с потрепанными приложениями, написанными бородатыми хиппи в 60-х. Нет, вы можете работать с самыми свежими приложениями, созданными бородатыми хиппи XXI века! В любом случае, FTP – это старомодный FTP-клиент; вам нужен *NCFTP*.

Строка состояния, завершение файловых имен, автовозобновление закачки, перенаправление через прокси – не соглашайтесь на меньшее, чем наличие всех функций, полагающихся приложению рабочего стола. *NCFTP* находится в разработке с 1991 года, и работает также на Windows и OS X, так что может оказаться очень полезным инструментом, с которым стоит познакомиться. В самом последнем релизе исправлено несколько ошибок и включено несколько обходных путей для разных типов серверов, которые не слишком идеально работают. Как-то так вышло, что те же разработчики заодно создают и *NCFTPD* – полнофункциональный сервер. LXF



› **Передача файлов в оболочке вовсе не должна быть морокой.**

Также вышли...

Новое и обновленное ПО, также заслуживающее внимания…

Flactagger 3.1.1
Типа тэгов Think ID3, но для аудио файлов FLAC. Эта удобная утилита может даже автоматически создавать тэги для многих файлов.
http://flactagger.berlios.de

SmartSVN 4.0.3
Обновление популярного клиента SVN.
www.syntevo.com/smartsvn

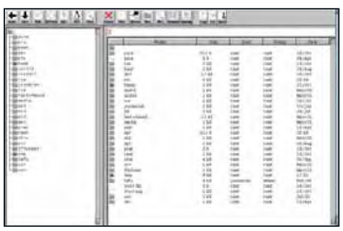
EndeavourII 2.10
Управление файлами никогда еще не было настолько простым. Снова!
http://freshmeat.net/projects/endeavour2

PiggyMarketSqueek 0.6.0
Следите за своими акциями и ценными бумагами, и вовремя получайте предупреждение, если что-то пойдет не так!
www.gthoreton.co.uk

GBSplay 0.0.91
Оживите незабвенные дни пищалок и квакалок с помощью этого музыкального плейера Nintendo Gameboy.
http://gbsplay.berlios.de

Clonezilla 0.3.2
Live-дистрибутив построенный вокруг образов разделов и восстановления системы.
http://clonezilla.sourceforge.net

Usermin 1.360
Основанное на web администрирование для обычного пользователя.
www.usermin.com



Xclip 0.1.1
Интерфейс командной строки для буфера обмена X Window System.
http://freshmeat.net/projects/xclip

Gambas 2.8.0
BASIC-подобный язык, действительно очень хороший.
http://gambas.sourceforge.net

Магазин-салон

НОВАЯ ТЕХНИЧЕСКАЯ КНИГА

190005, Санкт-Петербург,
Измайловский пр., 29

В МАГАЗИНЕ ПРЕДСТАВЛЕНА ЛИТЕРАТУРА по

компьютерным технологиям
радиотехнике и электронике
физике и математике
строительству и архитектуре
транспорту
машиностроению

и другим
естественно-научным
и техническим направлениям

Магазин работает
с 10.00 до 20.00
без обеденного перерыва
выходной день – воскресенье

✓ низкие цены ✓ прямые поставки от издательств ✓ ежедневное пополнение ассортимента ✓ подарки и скидки покупателям

Тел.: (812) 251-41-10,
e-mail: trade@techkniga.com
www.techkniga.com

LXF DVD111

Превратите свой ПК в медиацентр и испробуйте новые технологии Linux!

Забудьте весь этот «год Linux для настольных компьютеров»! Пора задавать другой вопрос: сейчас год Linux на вашем телевизоре? Благодаря Mythbuntu, который загружается прямо с **LXFDVD**, вы сможете создать домашний развлекательный центр, не тратя долгих часов на утомительную настройку. И выглядит он просто фантастически – как же далеко мы ушли от времен, когда Linux считался гадким утенком в стае компьютерных технологий!.

Майк Сондерс
Редактор диска
mike.saunders@futurenet.co.uk

Эмулятор ПК

VirtualBox

Мы в редакции **LXF** просто сами не свои до дистрибутивов, и, судя по разговорам на наших форумах, вы их тоже любите. Многие проекты сегодня выпускают Live-версии, что значительно упрощает знакомство с ними, поскольку не нужно ломать голову над разбиением жесткого диска; но у Live-дистрибутивов есть и своя теневая сторона. Самый заметный недостаток – при запуске с оптического носителя их производительность значительно уступает дистрибутивам, установленным на жесткий диск. Поэтому идеальным решением будет использование эмуляторов ПК – они позволяют работать с дистрибутивом, используя образы файлов на винчестере, и, следовательно, скорость будет почти как у установленного дистрибутива.

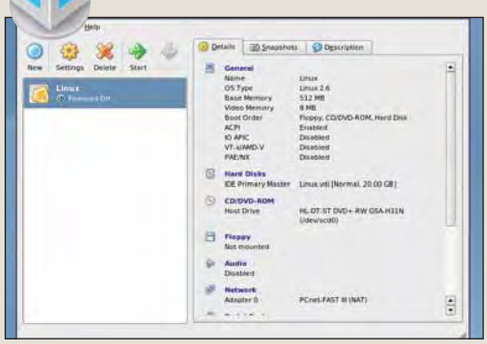
В этом месяце мы включили ISO-образы CD предварительных релизов Debian, Fedora и OpenSUSE (более подробную информацию см. на стр. 120). Используя *VirtualBox*, превосходный и свободный эмулятор ПК, не так давно приобретенный Sun Microsystems, вы сможете попробовать их, не опасаясь поломать свою систему. Если вы хотя бы когда-нибудь пытались запустить *Bochs* или *Qemu* и в итоге уронили голову на руки, не в силах одолеть опции командной строки, задействованные в этом процессе, то *VirtualBox* станет для вас глотком свежего воздуха с ароматом GUI.

Эмуляция оборудования

VirtualBox эмулирует все оборудование, имеющееся на обычном ПК: видео- и аудиокарты, жесткий диск, CD/DVD-приводы и пр. Любая ОС, пригодная для установки и запуска на настоящем компьютере, должна без проблем работать в *VirtualBox*. Вы можете использовать устройства USB, а затем эмулировать оборудование, которого, возможно, даже и нет на вашем ПК – например, последовательный порт.

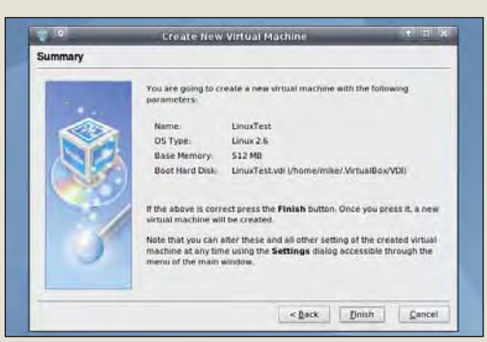
Несомненно, лучшая функция *VirtualBox* – поддержка моментальных снимков системы. В любой момент работы с ОС в виртуаль-

Шаг за шагом: Создание виртуальной машины в VirtualBox

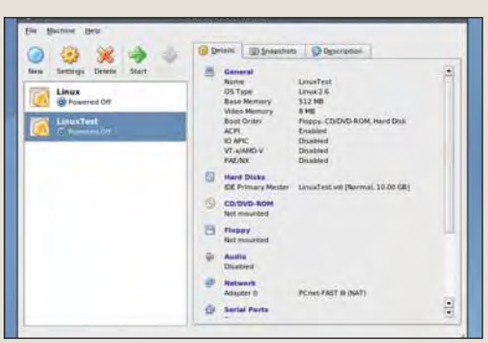


1 Запуск
Наберите **VirtualBox** в командной строке, и перед вами появится это окно. Мы создаем новый эмулированный ПК для ознакомления с дистрибутивами Linux, так что кликните по кнопке Создать вверху слева.

2 Имя
Задайте имя для своей виртуальной машины (например, LinuxTest), и затем выберите тип ОС во всплывающем списке. Для тестирования современных дистрибутивов подойдет пункт «Linux 2.6».



5 Подтверждение
После этого вам предложат подтвердить свое желание создать виртуальную машину – нажмите на кнопку Назад, если хотите что-то изменить.



6 Настройки
По окончании процесса вы попадете в панель управления, где можно пощелкать по синим ссылкам справа, чтобы произвести тонкую настройку.

ной машине вы можете приостановить эмуляцию и сохранить состояние на жестком диске. Это особенно полезно, если вы затеваете некие революционные преобразования, способные повредить установленным программам. Допустим, у вас в *VirtualBox* работает FooLinux 3.0, и вы хотите начать использовать его тестовые релизы. Прежде чем приступить к процедуре обновления, сделайте моментальный снимок в *VirtualBox*: если что-то пойдет не так, вы всегда сможете вернуться к более раннему состоянию. А еще это исключительно полезно для тех, кто сам балуется разработкой ОС.

Приступаем к виртуализации

Вы найдете *VirtualBox* в разделе Система нашего DVD. Если у вас Ubuntu 8.04, Fedora 9, Mandriva 2008, OpenSUSE 10.3 или Debian 4.0 (или более новые версии этих дистрибутивов), вы можете установить двоичные пакеты с диска. Например, в Ubuntu скопируйте соответствующий Deb-пакет для своей архитектуры в домашнюю директорию, откройте *Gnome Terminal* и введите:

```
dpkg -i virtualbox*.deb
```

Для систем на основе RPM (OpenSUSE, Fedora и Mandriva), скопируйте RPM в домашний каталог, откройте окно терминала, переключитесь в root с помощью **su** и введите

```
rpm -i VirtualBox*.rpm.
```

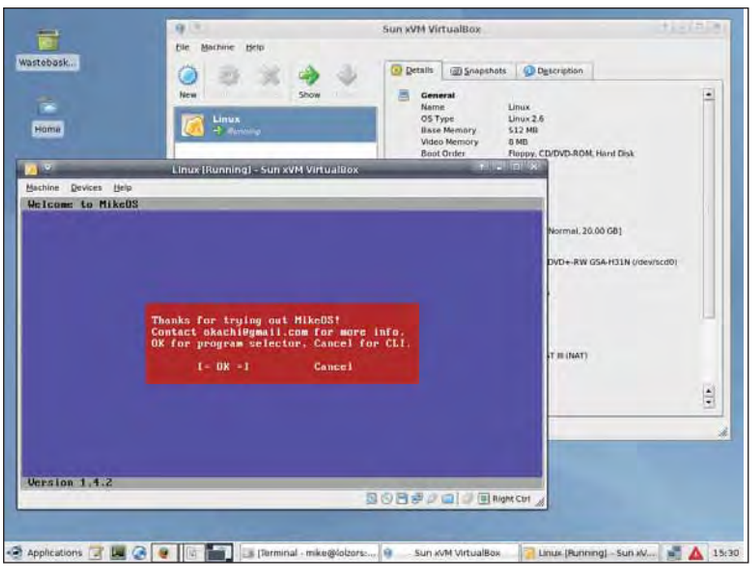
(Звездочка – это символ подстановки, он замещается именем файла.). Если вы работаете в другом дистрибутиве, скопируйте один из RUN-файлов с DVD в домашнюю директорию и введите **sh VirtualBox*.run** от имени root, чтобы установить программу. В случае проблем загляните в раздел **Справка/Новичку** в Linux на DVD, там находится руководство по использованию командной строки и установке программ.

После инсталляции вам напомнят, что к системной группе 'vboxusers' надо добавить учетные записи ваших пользователей, чтобы они могли запускать виртуальную машину. Это можно сделать с помощью инструментов настройки вашего дистрибутива (данная процедура зависит от системы), или, в командной строке, переключитесь на root (**sudo bash** или **su**) и введите:

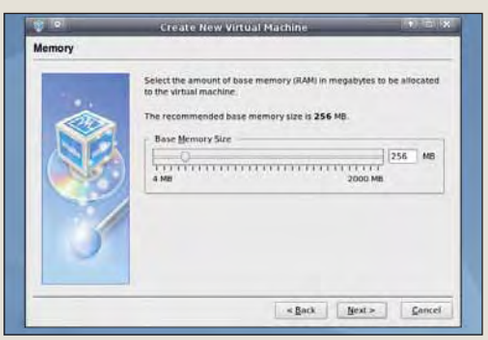
```
usermod -a -G vboxusers username
```

Естественно, вместо **username** надо указать имя своего пользователя. Теперь можно запустить эмулятор, введя '*VirtualBox*'; для настройки виртуальной машины следуйте приведенным внизу инструкциям.

По бессмертным словам Строителя Боба – да, можем (исправить то, что не так) [Песня: Bob the Builder, Can we fix it? Bob the Builder – Yes, we can. – Строитель Боб, это можно поправить? Строитель Боб – Да, можно, – прим. пер.] В случае проблем при работе с *VirtualBox* просмотрите список часто задаваемых вопросов на сайте проекта, **www.virtualbox.org/wiki/User_FAQ**. Если это не поможет, оставьте сообщение на форумах по адресу **http://forums.virtualbox.org** – не забудьте указать, с какой версией *VirtualBox* вы работаете (на **LXFDVD** – 2.0.2), какой у вас дистрибутив, и какую ОС вы пытаетесь запустить на виртуальной машине. Кто-нибудь из постоянных участников форума укажет вам верный путь. »

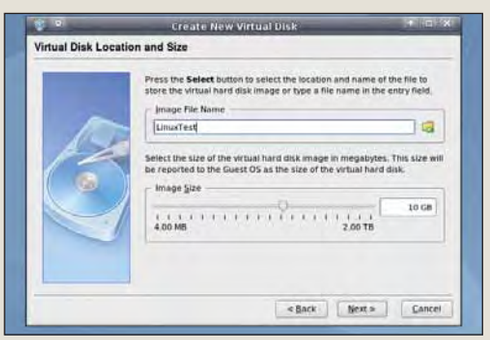


› ***VirtualBox*** **способен работать со всеми видами ОС для x86, от крупных решений уровня предприятия до крошечных проектов, умещающихся на дискетке.**



3 Память

Затем выделите память для виртуальной машины. Больше половины вашего реального ОЗУ выделять не стоит: это может затормозить установленную систему. Чтобы испытывать дистрибутивы, потребуется 512 МБ.

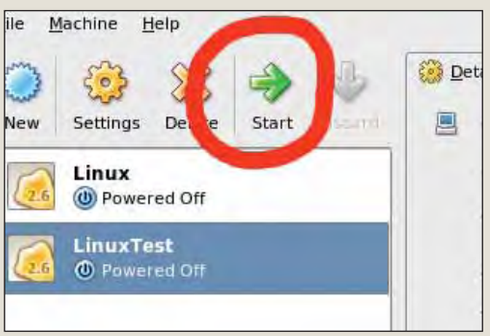


4 Жесткий диск

Теперь можно создать образ виртуального жесткого диска. Для общего тестирования дистрибутивов вам не понадобится более 10 ГБ. Выберите опцию «динамического расширения», чтобы сэкономить немного места на винчестере.

7 CD/DVD

Чтобы использовать ISO-образ CD, например, образ диска дистрибутива, скопированный с **LXFDVD**, нажмите на CD/DVD-ROM на панели управления, выберите Подключить CD/DVD, затем укажите нужный ISO-файл.



8 Поехали!

И, наконец, нажмите на Старт вверху слева в основном окне, для запуска виртуальной машины и ее загрузки с ISO-образа. Чтобы вновь обрести контроль над мышью, нажмите правую клавишу Ctrl.

Дистрибутивы Linux

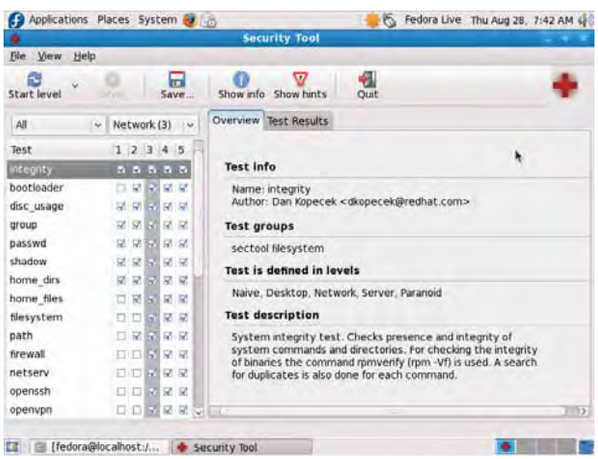
Предварительные релизы

Разработка дистрибутивов сейчас кипит вовсю, и в этом месяце мы попробуем кое-что новенькое: на нашем диске – три пробных релиза Fedora, Debian и OpenSUSE. Они пока находятся в стадии разработки, но уже имеют несколько потрясающих новых функций, достойных ознакомления.

Эти дистрибутивы идут на **LXFDVD** (см. раздел Дистрибутивы) в формате ISO-образов CD. Вы можете записать их на CD-R и запустить на настоящем ПК, но есть путь гораздо проще – без прожига: установить *VirtualBox* и загрузить ISO-образы в виртуальной машине. Помните, что это релизы находятся на промежуточной стадии своего развития, и они вряд ли свободны от ошибок и недоделок. Поэтому мы не рекомендуем устанавливать их на настоящий ПК, если только вы не хотите присоединиться к процессу разработки. И, как всегда, обязательно создавайте резервные копии важной информации!

> «Это — ранние релизы, с суперскими функциями, достойными знакомства.»

И, наконец, у нас есть Fedora 10 Beta. Чтобы испытать этот дистрибутив, понадобится CPU с поддержкой виртуализации: AMD Athlon 64, 2-е поколение Opteron или Phenom с AMD-V; или определенные версии чипов Intel Core Due или Core 2 Duo с VT-x. Узнать, поддерживает ли ваш CPU эти технологии, можно, перезагрузив свой настоящий ПК, зайдя в настройки BIOS и проверив, есть ли опция виртуализации среди настроек. Если есть, включите ее и перезагрузитесь. Затем в *VirtualBox* нажмите на кнопку Свойства, перейдите на вкладку Дополнительно, и убедитесь, что флажок Включить VT-x/AMD-v установлен.



› ***SecTool*** **надеется сделать настройку безопасности Fedora 10 более мощной и понятной.**

Отпадное трио

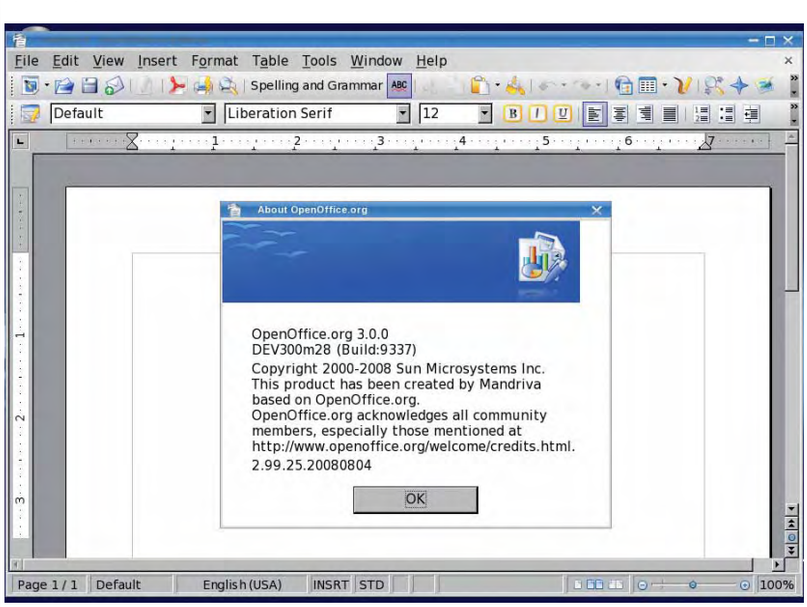
Debian славен своим умением затягивать релиз-цикл, и версия 5.0, известная как Lenny, увы, не стала исключением. К счастью, ISO-образы «слепка» тестируемого дистрибутива генерируются еженедельно, и вы всегда можете быть в курсе последних новостей. Стабильности Etch от него, разумеется, ожидать пока не следует, зато пакеты новее. Если удастся отловить какую-нибудь странную ошибку (и вы уверены, что она не вызвана *VirtualBox*) – не забудьте сообщить о ней разработчикам.

Для OpenSUSE 11.1 Beta 3, загрузите ISO в *VirtualBox* и нажмите в появившемся меню F3, чтобы выбрать видео режим VESA. Затем введите **noreplace-paravirt** в строке внизу и нажмите Enter. С момента выхода 11.0 изменилось немногое, но самое крупное обновление – переход с KDE 4.0 на 4.1.

Самые радикальные изменения по сравнению с Fedora 9 – Gnome 2.23.5, графическая система загрузки под названием *Plymouth* и новая среда безопасности. SecTool объединяет в себе системный аудит и обнаружение вторжений – для его установки откройте окно терминала, наберите «su», чтобы переключиться на пользователя root, и введите:

```
yum install sectool sectool-gui
```

Когда *Yum* завершит установку SecTool, введите **sectool-gui**, чтобы его запустить.



› **Mandriva 2009 и *OpenOffice.org 3.0* были тестовыми версиями, когда мы это писали, но к моменту получения журнала вами уже будут доступны финальные релизы.**

Очень ВАЖНО!

Перед тем, как вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим:

Диски *Linux Format* DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция *Linux Format* не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензии.

Бракованные диски

В маловероятном случае обнаружения бракованного диска *Linux Format*, просим связаться с нашей группой поддержки по адресу disks@linuxformat.ru для получения содействия.

ВНИМАТЕЛЬНО ПРОЧТИТЕ ЭТО ПЕРЕД ТЕМ, КАК ИСПОЛЬЗОВАТЬ DVD-ДИСК !

Информация о диске

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, следует заглянуть именно туда.

ФОРМАТЫ ПАКЕТОВ

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любые другие. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными бинарными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы смогли собрать его самостоятельно.

ДОКУМЕНТАЦИЯ

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

ЧТО ЭТО ЗА ФАЙЛЫ?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux, различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

имя_программы-1.0.1.i386.rpm – вероятно, это бинарный пакет RPM, предназначенный для работы на системах x86;

имя_программы-1.0.1.i386.deb – такой же пакет, но уже для Debian;

имя_программы-1.0.1.tar.gz – обычно это исходный код;

имя_программы-1.0.1.tgz – тот же файл, что и выше по списку: «tgz» - это сокращение от «tar.gz»;

имя_программы-1.0.1.tar.bz2 – тот же файл, но сжатый bzip2 вместо обычного gzip;

имя_программы-1.0.1.src.rpm – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

имя_программы-1.0.1.i386.FC4.RPM – бинарный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

имя_программы-1.0.1.ppc.Suse9.rpm – бинарный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

имя_программы-devel-1.0.1.i386.rpm – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Содержание LINUX FORMAT DVD Ноябрь2008

Сторона 1

Аудио
Audacity - продвинутый аудиоредактор
Decibel - аудиоплеер на GTK+
LMMS - альтернатива FruityLoops

Рабочий стол
Basket - программа для ведения небольших заметок
Chandler - открытый персональный органайзер
gDesklets - инструмент для отображения десклетов
GTD-Free - персональный TODO-менеджер
KDE - популярное рабочее окружение
Kontact - многофункциональный органайзер для KDE
MonkeyGTD - GTD-система, основанная на TiddlyWiki и MPTW
Org-mode - режим редактора Emacs для ведения заметок
R_GUI - графические интерфейсы для R
Thinkig Rock - приложение для сбора мыслей по технологии GTD
Tracks - веб-приложение, реализующее методологию GTD

Разработка
Bless – высококачественный полнофункциональный hex-редактор
Gambas - графическая среда разработки
GTK-server - поток-ориентированный интерфейс для библиотек GTK
Little Wizard - среда разработки для детей
Qt - инструментарий для создания кроссплатформенных программ

Дистрибутивы
AntiX - "легкая" версия Mepis Linux
Debian - последняя еженедельная сборка Lenny
Fedora Live - бета-версия "живой" Fedora
Haiku - открытый клон BeOS
Mythbuntu - специальная версия Ubuntu для запуска MythTV
OpenSUSE Live - "живая" версия OpenSUSE

Игры
Eternal Lands - свободная MMORPG
Holotzs_Castle - открытый платформер
SDL-Ball - клон арканоида
SDL Asylum - порт компьютерной игры Asylum
The Mana World - открытая MMORPG на базе eAthena

Hot_Picks
Aria2 - утилита для скачивания файлов
Brain Workshop - открытая реализация Dual N-Back
Diet Monger Ass Kicker - программа для составления диет
GNUjump - клон простой игрушки Xjump
KleanSweep - приложение для удаления ненужных файлов
Luke-SDK - интегрированная среда разработки
NcFTP - комплект свободных программ для работы с протоколом FTP
Urssus - RSS-агрегатор на PyQT

Графика
Camera Life - основанный на веб-технологиях каталогизатор изображений
Digikam – продвинутое приложение для управления фотоколлекцией
pfstools - пакет утилит для работы с HDR

Справка
Ответы на часто задаваемые вопросы
Новичку в Linux
RUTE - книга по администрированию GNU/Linux

Интернет
GCALDaemon - утилита для синхронизации календарей с сервисами Google
Lightning - календарь для Thunderbird
Opera - набор приложений для Интернета
Sunbird - календарь-органайзер
Thunderbird - почтовый клиент

Сервер
Icecast - потоковый аудиосервер
NuFW - сетевой фильтр

Система
Bonnie++ - утилита для тестирования производительности жёстких дисков
Furius - интерфейс для работы с образами дисков
Nagios - сервис мониторинга сети
VirtualBox - виртуальная машина

Сторона 2

Alt Linux Desktop 4.1

Комментарии? Присылайте ваши мысли и предложения по электронной почте: info@linuxformat.ru

Пожалуйста, ознакомьтесь с опубликованной в журнале инструкцией перед использованием данного диска.

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не может принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Дефектные диски. В маловероятном случае обнаружения дефектов на данном диске, пожалуйста, обращайтесь по адресу: disks@linuxformat.ru

Тираж изготовлен ООО «Фортмедиа», Россия, Санкт-Петербург, 196006 ул. Цветочная д. 7, тел. +7 (812) 388-8290. Лицензия МПТР России ВАФ № 77-225

Поставляется вместе с журналом LINUXFORMAT номер 11 (111) Ноябрь 2008

DVD

СОЗДАНИЕ УСТАНОВОЧНЫХ ДИСКОВ ПРИ ПОМОЩИ CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Для начала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем, после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь вы можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust!. Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что, если у вас нет устройства, с помощью которого можно было записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков, подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика. LXF

Новые программы

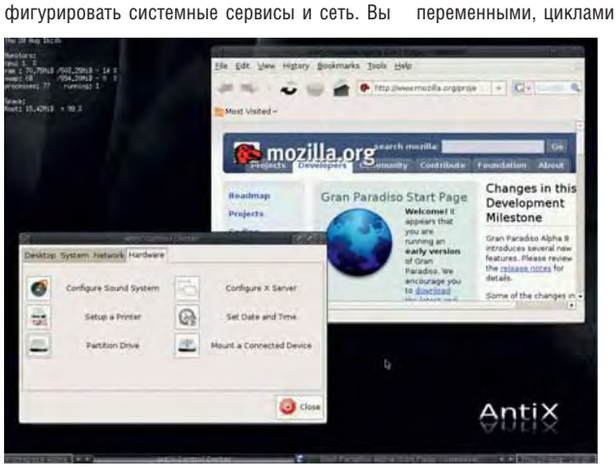
Antix, Opera и средства разработки

Небось, успели подумать, что тут и конец веселью насчет дистрибутивов? А мы припрятали на нашем DVD еще один. Помните Mepis, проект Уоррена Вудфорда [Warren Woodford], который блистал в материале LXF79? Так вот, *Antix* (**http://antix.mepis.org**) – это легкий дистрибутив на базе Mepis, задуманный как быстрая и дружелюбная ОС для старых машин. Вы можете запускать его на 266-МГц (x86) машине с 64 МБ ОЗУ – но без особой многозадачности. Для более удобной работы рекомендуется 128 МБ ОЗУ. Дистрибутив содержится на **LXFDVD** в виде ISO-образа CD для записи на CD-R.

В окне загрузки *Antix* вы увидите массу опций: выберите загрузку по умолчанию для обычного режима Live CD; Lite-NoNet, если вам нужна версия с меньшим потреблением памяти; и Vesa, если у вас проблемы с графикой (или если вы работаете в *VirtualBox*). В окне приглашения вы увидите поля для ввода имени пользователя и пароля – наберите demo в качестве того и другого. Через пару секунд перед вами появится рабочий стол; *Antix* использует оконный менеджер *Fluxbox*, который управляется с вашими приложениями быстрее, чем Аршавин с мячом.

Это Antix

Щелкните правой кнопкой мыши на рабочем столе, чтобы вывести меню запуска программ; из него можно запустить web-браузер *Iceweasel* (тот же *Firefox*, но без каких-либо проблем с товарными знаками), клиент электронной почты *Claws* и текстовый редактор *AbiWord*. В меню программ вы найдете также несколько медиаплейеров и игр. Для настройки дистрибутива, выберите Настройки > Центр управления [Settings > Control Centre]: это позволит изменить разрешение экрана, сконфигурировать системные сервисы и сеть. Вы также найдете здесь *GParted*, отличный редактор дисковых разделов, который придет на помощь в случае проблем с винчестером. Если вы захотите установить *Antix* на жесткий диск, выберите в главном меню пункт Install. Помните: пароль для root – просто root.



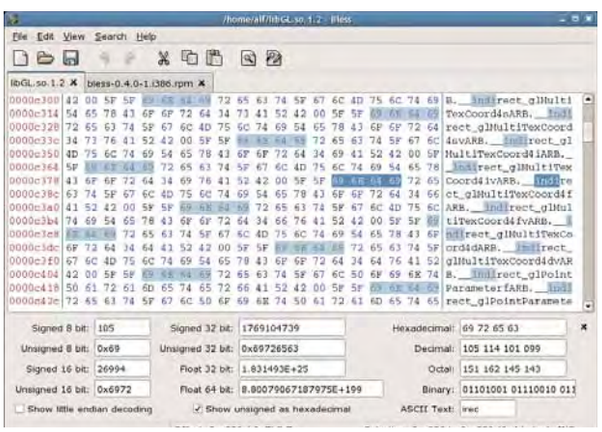
› Стильная черная тема оформления *Antix* иллюстрирует скорость дистрибутива – он невероятно быстро работает на современных машинах.

Время кодировать

Впечатлены нашим обзором *Opera* на стр. 15 и хотите попробовать сами? Вы найдете этот браузер на **LXFDVD** в разделе **Интернет**. *Opera Link* (LXF110) в этом релизе обзавелся новыми возможностями и теперь позволяет синхронизировать поисковые движки и историю посещений. Если, находясь в офисе, вы нашли какой-то интересный сайт, а рабочий день закончился так внезапно, не нужно добавлять ресурс в закладки: домашней браузер будет в курсе похождений хозяина и без проблем позволит вам продолжить чтение. Попутно вы отметите, что *Opera*, и до того не считавшаяся медлительной, стала еще быстрее – в том числе и благодаря наличию нового режима минимизации трафика в почтовом клиенте *Opera Mail*. Эту функцию по достоинству оценят те, кто часто пользуется GPRS-соединением.

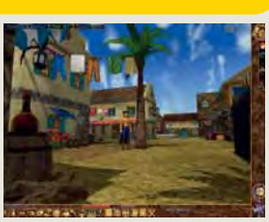
В разделе **Разработка** мы предлагаем вашему вниманию несколько симпатичных инструментов для программирования. Если вы планируете собирать окружение KDE 4.1 из исходных текстов (см. **Рабочий стол**), вам пригодятся библиотеки *Qt*, от которых оно зависит. Понятно, что компиляция программного пакета вроде KDE, достойного Гаргантюа, требует немалого труда, так почему бы не призвать на помощь лилипутов? *Little Wizard* – среда разработки на основе иконок, чья цель – сделать программирование настоящим удовольствием, понятным даже для малышни. Используя одну лишь мышь, дети могут разобраться с переменными, циклами и прочими аспектами программирования.

У нас есть еще *Gambas* – IDE, базирующаяся на интерпретаторе Basic. Некоторые кодеры могут заявить, что Basic – не лучший выбор для написания приложений, поскольку построенный им код напоминает мешанину спагетти, и его трудно поддерживать. Но это отнюдь не старый Basic в стиле **10 PRINT "HELLO"** из эпохи Speccy – это структурированный модульный язык, вполне пригодный для быстрого написания графических утилит. И, наконец, стоит упомянуть *Bless*, шестнадцатеричный редактор, написанный на C# с помощью Mono. Всегда полезно иметь под рукой шестнадцатеричный редактор на случай, если потребуется исправить поврежденный файл, а *Bless* – один из лучших из



› *Bless* предоставляет солидную цифровую информацию для быстрого шестнадцатеричного редактирования или внесения изменений в бинарники.

И наконец... Игры, ау!



› *Eternal Lands*: единороги, драконы и прачечные.

В этом месяце увидели обновление несколько очень многообещающих ролевых игр (RPG), так что – полный вперед, в раздел Игры нашего диска, рассмотрим их поподробнее. *Eternal Lands* – серьезная попытка создать 3D MMORPG для Linux – то есть, massively multiplayer online RPG, многопользовательскую RPG. Несмотря на то, что она пока на бета-стадии, в нее уже вполне можно играть, и она содержит элементы, которые и следует ожидать от RPG: сражения с монстрами, исследование новых территорий и развитие навыков. Смотрите файл **install.txt** на DVD, там имеются подсказки по запуску игры, и директорию help с информацией по управлению интерфейсом и вводу команд.

В свою очередь, *Mana World* визуально представляет собой нечто более традиционное, с двумерной графикой, обеспечивающей вид сверху, и персонажами из мультиков. Внешне она воспроизводит классические 8/16-битные RPG, вроде *Final Fantasy* и *Zelda*, но она еще тоже на ранней стадии разработки. И если вам всегда хотелось поучаствовать в игровом проекте – быть кодером, тестером, художником, музыкантом или создателем документации – зайдите на сайт **www.themanaworld.org** и предложите им руку помощи. Это будет очень благодарный труд.

И еще у нас есть *SDL Asylum*, изумительно изобретательная игра-платформер, основанная на классике – *Acorn Archimedes*. Ваша задача – провести Зигмунда [Sigmund] по множеству фрейдистских уровней (Ego, Psyche и Id), истребляя «нейроны-ренегаты», чтобы излечить умственное расстройство несчастного мальчика. А если вы не любитель чокнутых, попробуйте *Holotz's Castle*, отличную игрушку, в которой вам придется изыскивать способы сбежать из замка.

Журнал Linux Format начинает подписную компанию 2009!

Подпишись на 12 номеров журнала Linux Format на 2009 год и выиграй ноутбук ASUS или интернет-таблет Nokia или один из множества призов !!!

Linux Format - ежемесячный журнал, посвященный исключительно Linux и Open Source. Объем журнала - 114 полос, кроме того, к каждому номеру прилагается DVD-диск, содержащий популярные дистрибутивы Linux/BSD, объемные пакеты и просто интересное ПО.

Номера отправляются ценными бандеролями. Кроме того, принимаются подписки на полугодие и продаются отдельные номера.

Оформите подписку на сайте : www.linuxcenter.ru

Подписку можно оформить в агентствах Роспечать и Пресса России

Подписные индексы : "Роспечать" - 20882, "Пресса России" - 87974

ПОДПИСКА НА LINUX FORMAT

ПОДПИСКА В ЛИНУКСЦЕНТРЕ

Сколько стоит подписка?

Подписка на журнал «Linux Format» 12 номеров (январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь) стоит 1800 рублей
Подписка на журнал «Linux Format» 6 номеров (июль, август, сентябрь, октябрь, ноябрь, декабрь 2009 года) стоит 900 рублей

Как оформить подписку?

Чтобы оформить подписку на журнал «Linux Format», необходимо зарегистрироваться в интернет-магазине Linuxcenter.Ru, указав ФИО и подробный почтовый адрес подписчика, заказать товар «Подписка на журнал «Linux Format» 12 номеров 2008 года», или товар «Подписка на журнал «Linux Format» второе полугодие 2008 года», получить от системы квитанцию для оплаты в любом отделении Сбербанка (для физических лиц) или счет для оплаты по безналичному расчету (для юридических лиц)

Как оплатить подписку?

– по выставленному счету (для юридических лиц)
– по квитанции в любом отделении Сбербанка

Плюсы подписки

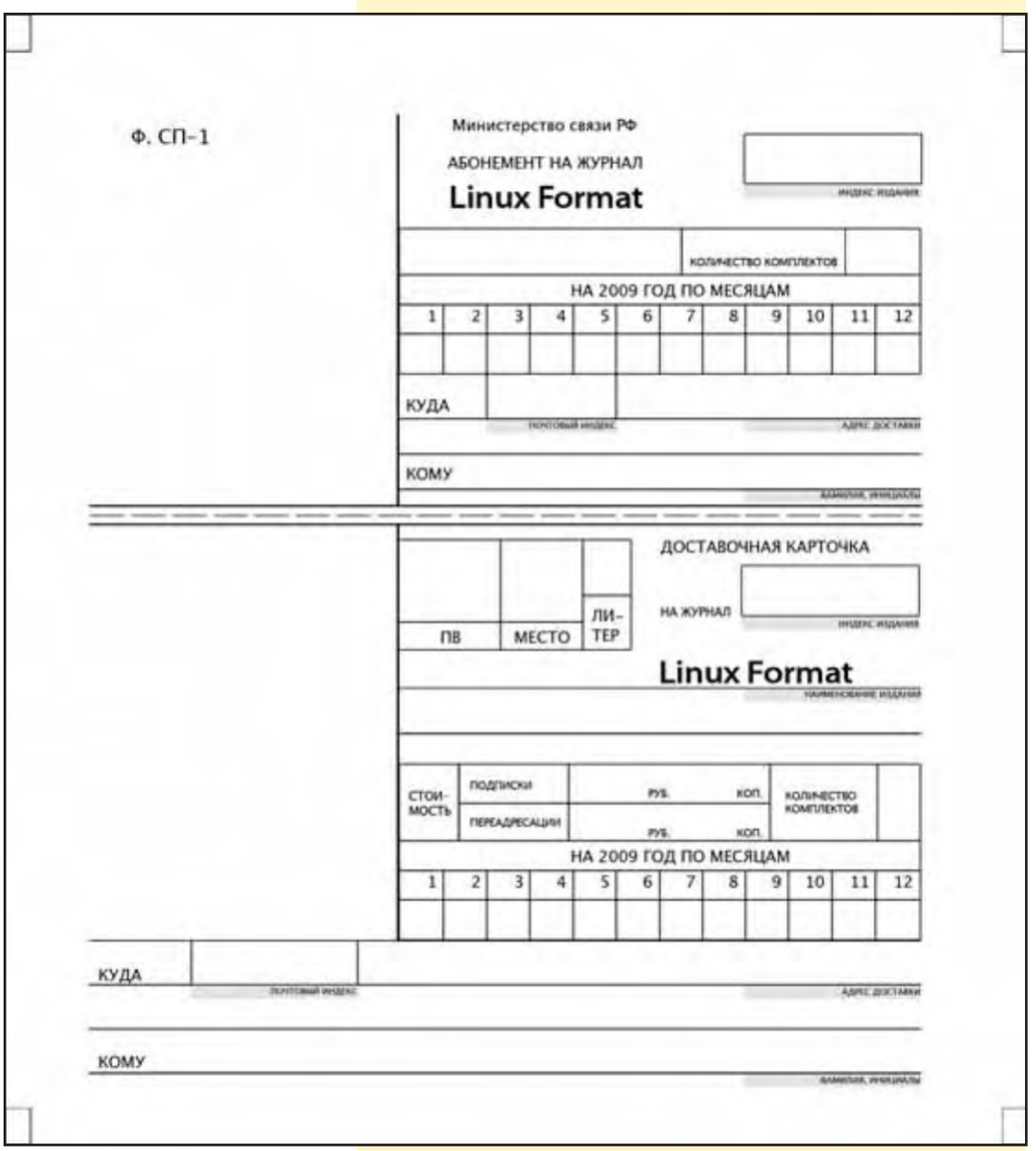
- подписка дешевле!
- гарантированное получение нового номера журнала!

ПОДПИСКА - 2009!

ПОДПИСКА ПО КАТАЛОГАМ РФ

Каталог агентства «РОСПЕЧАТЬ» – подписной индекс 20882

Каталог «ПРЕССА РОССИИ» – подписной индекс 87974


Ф. СП-1

Министерство связи РФ

АБОНЕМЕНТ НА ЖУРНАЛ

Linux Format

ИНДЕКС ИЗДАНИЯ

КОЛИЧЕСТВО КОМПЛЕКТОВ

НА 2009 ГОД ПО МЕСЯЦАМ

1	2	3	4	5	6	7	8	9	10	11	12

КУДА

ПОЧТОВЫЙ ИНДЕКС

АДРЕС ДОСТАВКИ

КОМУ

ФАМИЛИЯ, ИНИЦИАЛЫ

ДОСТАВОЧНАЯ КАРТОЧКА

НА ЖУРНАЛ

ИНДЕКС ИЗДАНИЯ

ПВ	МЕСТО	ЛИТЕР

Linux Format

НАИМЕНОВАНИЕ ИЗДАНИЯ

СТОИМОСТЬ	ПОДПИСКИ	РУБ.	КОП.	КОЛИЧЕСТВО КОМПЛЕКТОВ
	ПЕРЕАДРЕСАЦИИ	РУБ.	КОП.	

НА 2009 ГОД ПО МЕСЯЦАМ

1	2	3	4	5	6	7	8	9	10	11	12

КУДА

ПОЧТОВЫЙ ИНДЕКС

АДРЕС ДОСТАВКИ

КОМУ

ФАМИЛИЯ, ИНИЦИАЛЫ

ПОДПИСКА НА LINUX FORMAT

ПОДПИСКА ПО КАТАЛОГАМ СНГ И БЛИЖНЕГО ЗАРУБЕЖЬЯ

Каталог «Российская Пресса» – совместный проект Государственного предприятия «Казпочта», Агентства «Книга-Сервис» и АРЗИ.

Блок изданий АРЗИ в национальных Каталогах Украины и Беларуси. В Азербайджане, Армении, Грузии, Киргизии, Узбекистане и Молдове - по изданиям, включенным в Объединенный каталог, распространяемые через АРЗИ.

Азербайджан
- по Объединенному каталогу российских изданий через Предприятие по распространению печати «Гасид»(370102, г. Баку, ул. Джавадхана, 21);

Армения
- по списку номенклатуры «АРЗИ» через ГЗАО «Армпечать» (375005, г.Ереван, пл.Сасунци Давида, д.2) и ЗАО «Контакт-Мамул» (375002, Г.Ереван, ул.Сарьяна, 22);

Белоруссия
- по Каталогу изданий стран СНГ через РГО «Белпочта» (220050, г.Минск, пр-т Ф.Скорины, 10);

Грузия
- по списку номенклатуры «АРЗИ»» через АО «Сакпресса» (380019, г.Тбилиси, ул.Хошараульская, 29) и АО «Мацне» (380060, г.Тбилиси, пр-т Гамсахурдия, 42);

Казахстан
- по Каталогу «Российская Пресса» через ОАО «Казпочта» и ЗАО «Евразия пресс»;

Молдавия
- по каталогу через ГП «Пошта Молдавей» (МД-2012, г.Кишинев, бул.Штефан чел Маре, 134);
- по списку через ГУП «Почта Приднестровья» (MD-3300, г.Тирасполь, ул.Ленина, 17);
- по прайс-листу через ООО Агентство «Editil Periodice» (2012, г.Кишинев, бул. Штефан чел Маре, 134).

Узбекистан
- по Каталогу «Davriy nashrlar» российские издания через Агентство по распространению печати «Davriy nashrlar» (7000029, Ташкент, пл.Мустакиллик, 5/3, офис 33);

Украина
- Киевский главпочтамт.
- Подписное агентство «KSS» Телефон/факс (044)270-62-20, 270-62-22

ПОДПИСКА НА LINUX FORMAT

Агентство "Centerpress"

Сколько стоит подписка?
Подписка на журнал "Linux Format" 12 номеров (январь, февраль, март, апрель, май, июнь, июль, август, сентябрь, октябрь, ноябрь, декабрь 2009 года) стоит 1800 рублей.

Как оформить подписку?
Чтобы оформить подписку на журнал "Linux Format", необходимо зарегистрироваться в интернет-агентстве Centerpress.ru, указав ФИО и подробный почтовый адрес подписчика, заказать товар «Подписка на журнал "Linux Format" на 2009 год 12 номеров (01-12/2008)», получить от системы квитанцию для оплаты в любом отделении Сбербанка (для физических лиц) или счет для оплаты по безналичному расчету (для юридических лиц)

Агентство "Centerpress": www.centerpress.ru

Все Плюсы подписки!
- Подписка дешевле!
- Гарантированное получение журнала!

По каталогам РФ

Каталог агентства "РОСПЕЧАТЬ" – подписной индекс
20882

Каталог "ПРЕССА РОССИИ" – подписной индекс
87974

АЛЬТЕРНАТИВНЫЕ АГЕНТСТВА РФ

Агентство «Интер-Почта»
(495) 500-00-60, курьерская доставка по Москве.

Агентство «Вся Пресса»
(495) 787-34-47

Агентство «УралПресс»

- **Екатеринбург, Березовский, В. Пышма, Первоуральск тел. (343) 375-80-71, 375-84-93, 375-84-39, факс 375-62-74, info@ural-press.ru**
- **Нижний Тагил тел. (3435) 411448, 417709, ntagil@ural-press.ru**
- **Челябинск тел. (351) 262-90-03, 262-90-05, pochta@chel.surnet.ru**
- **Пермь тел. (3422) 60-24-40, 60-22-95, 60-35-42, parma-press@permonline.ru**

Mandriva

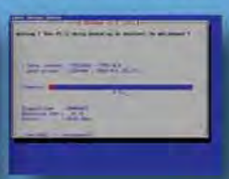
LINBOX RESCUE SERVER

Компания Мандрива.Ру
т/ф: +7-499-271-49-55
e-mail: info@mandriva.ru
web: www.mandriva.ru

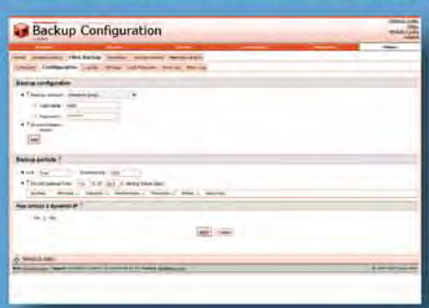
LINBOX RESCUE SERVER: полный контроль за ИТ-инфраструктурой

- Легкость администрирования за счет использования единой мультифункциональной Веб-консоли

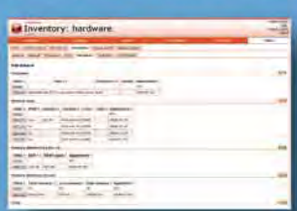
- Резервирование и восстановление систем с Windows или GNU/Linux за несколько минут



- Полный учет программного обеспечения и оборудования



- Резервирование и восстановление файлов пользователей как под Windows и GNU/Linux, так и под Unix



- Удаленный контроль за клиентскими ПК

Linbox Rescue Server (LRS) это пакет программ, предоставляющий все функции надлежащего локального или удаленного управления ИТ системой. Он может быть использован как в одном подразделении, так и в десятках, размещая, контролируя и исправляя все (в основном Windows и GNU/Linux) компьютеры в сетях LAN и WAN. LRS может установлен на машину с GNU/Linux с любой поддерживаемой локальной сетью. Позволяет управлять через Веб-браузер с любого локального или удаленного компьютера. Модуль резервного копирования использует систему загрузки PXE и не требует установки специализированного клиентского ПО.

+7 499 271 4955

ПОЗВОНИ И ПОЛУЧИ СВОБОДНУЮ ВЕРСИЮ LRS БЕСПЛАТНО

LINUX FORMAT
Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 6000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ:
ГЛАВНЫЙ РЕДАКТОР
Валентин Синицын info@linuxformat.ru
ВЫПУСКАЮЩИЙ РЕДАКТОР
Родион Водейко
Литературные редакторы
Елена Толстякова
Переводчики
Илья Аввакумов, Александр Бикмеев, Светлана Кривошеина, Александр Казанцев, Алексей Опарин, Валентин Развозжаев, Татьяна Цыганова
Редактор диска
Александр Кузьменков
Допечатная подготовка
Мария Пучкова, Родион Водейко, Семен Фрумкин
Креативный директор
Станислав Медведев
Технический директор
Денис Филиппов
Директор по рекламе
Денис Игнатов +7 812 965 7236 advert@linuxformat.ru
Генеральный директор
Павел Фролов
Заместитель генерального директора
Софья Виниченко
УЧРЕДИТЕЛИ
частные лица
ИЗДАТЕЛИ
Станислав Медведев, Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Текст», ООО «ППК «Текст»
188680, Ленинградская область, Всеволожский район, Колтуши, д.32
Заказ _____

Пре–пресс: *d.r.i.v.e-group*

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ:
Редактор Пол Хадсон [Paul Hudson] paul.hudson@futurenet.co.uk
Редактор обзоров Грэм Моррисон [Graham Morrison]
Редактор новостей Майк Сондерс [Mike Saunders] mike.saunders@futurenet.co.uk
graham.morrison@futurenet.co.uk
Художественный редактор Эфрейн Эрнандес-Мендоса (Efrain Hernandez-Mendoza) efrain.hernandez-mendoza@futurenet.co.uk
Литературные редакторы Эндрью Грегори [Andrew Gregory] agregory@futurenet.co.uk, Мэтт Нейлон [Matt Nailon] mnailon@futurenet.co.uk

Подготовка материалов
Нейл Ботвик [Neil Bothwick], д-р Крис Браун [Dr. Chris Brown], Энди Ченнел [Andy Channelle], Марко Фиоретти [Marco Fioretti], Майкл Дж. Хэммел [Michael J. Hammel], Джульетта Кемп [Juliet Kemp], Сьюзен Линтон [Susan Linton], Саймон Пиксток [Simon Pickstock], Алистер Рош [Alastor Roch], Ник Вейч [Nick Veitch], Евгений Балдин, Антон Борисов, Андрей Боровский, Анастасия Ильичева, Антон Коробейников, Евгений Крестников, Никита Лялин, Игорь Новиков, Галина Пожарина, Андрей Прахов, Алексей Радионов, Алексей Сиваков, Алексей Федорчук, Андрей Шевель, Василий Шингур
Художественные ассистенты: Стейси Блек [Stacey Black], Нейл Фристоун [Neil Freestone], Фил Хейкрэфт [Phil Haycraft], Кейт МакДоннелл [Kate McDonnell]
Иллюстрации: Крис Винн [Chris Winn], iStock Photo

КОНТАКТНАЯ ИНФОРМАЦИЯ
UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel 01225 442244 Email: linuxformat@futurenet.co.uk
РОССИЯ:
Санкт-Петербург (редакция): Лиговский пр. 50 корп. 15, тел. +7 (812) 640-49-90
Представительство в Москве:
м. Авиамоторная, Красноказарменная улица, дом 17. В помещении АТС МЭИ.
Тел/факс: +7(499) 271-49-54.
По вопросам сотрудничества, партнерства, оптовых закупок partner@linuxcenter.ru

Авторские права: Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензией Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

LINUX – зарегистрированный товарный знак Линуса Торвальдса [Linux Torvalds]. «GNU/Linux» заменяется на «Linux» в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

За информацией о журналах, издаваемых Future plc group company, обращайтесь
http://www.futureplc.com

© Linux Format 2005
© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO •SAN FRANCISCO

в декабре

Будьте креативны!

Художники, web-дизайнеры, создатели фильмов и прочие творческие личности всех мастей, объединяйтесь! Вам нечего терять, кроме своих лицензионных цепей.

Решаем проблемы KDE4

Если новейший свободный рабочий стол сбивает вас с толку, не пугайтесь – вы не одиноки. Мы сами запутались настолько, что пришлось писать руководство по выходу из тьмы к свету.

Смени дистрибутив!

Вам все-таки надоел коричневый или вы прониклись идеями Hurd? Неважно – мы научим, как сменить систему с наименьшей головной болью.

Управление логическими томами

Стать гуру виртуальных разделов может каждый!

Содержание последующих выпусков может меняться без уведомления